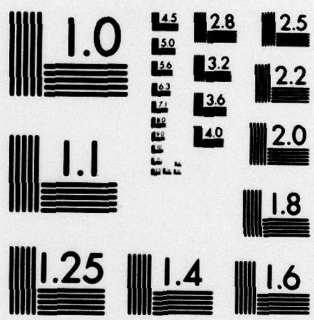
AD-A073 587 NAVAL AIR DEVELOPMENT CENTER WARMINSTER PA AIR VEHICL--ETC F/G 1/2
A COMPUTERIZED VSTOL/SMALL PLATFORM LANDING DYNAMICS INVESTIGAT--ETC(U)
SEP 77 R L NAVE

UNCLASSIFIED NADC-77024-30

1 OF 4
AD
A073587


1.0
4.5
5.0
5.6
6.3
7.1
8.0
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.1
1.8
1.25
1.4
1.6

REPORT NO. (14) NADC-77024-30

(2) B.S.

LEVEL II

6224110

AD A073587

(6)

A COMPUTERIZED VSTOL/SMALL PLATFORM LANDING DYNAMICS INVESTIGATION MODEL

(10) Ronald L. Nave
Air Vehicle Technology Department
NAVAL AIR DEVELOPMENT CENTER
Warminster, Pennsylvania 18974

(12) 236p.

(11) 30 SEP[redacted]77

DDC
RECEIVED
SEP 11 1979
A

(9) FINAL REPORT

AIRTASK NO. WF41-421-203
Work Unit No. MX2X1

DDC FILE COPY

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

(16) F41421

(17) WF41421203

Prepared for
NAVAL AIR SYSTEMS COMMAND
Department of the Navy
Washington, D.C. 20361

407207

79 09 10 052

NOTICES

REPORT NUMBERING SYSTEM - The numbering of technical project reports issued by the Naval Air Development Center is arranged for specific identification purposes. Each number consists of the Center acronym, the calendar year in which the number was assigned, the sequence number of the report within the specific calendar year, and the official 2-digit correspondence code of the Command Office or the Functional Department responsible for the report. For example: Report No. NADC-76015-40 indicates the fifteenth Center report for the year 1976, and prepared by the Crew Systems Department. The numerical codes are as follows:

CODE	OFFICE OR DEPARTMENT
00	Commander, Naval Air Development Center
01	Technical Director, Naval Air Development Center
02	Program and Financial Management Department
09	Technology Management Office
10	Naval Air Facility, Warminster
20	Aero Electronic Technology Department
30	Air Vehicle Technology Department
40	Crew Systems Department
50	Systems Department
60	Naval Navigation Laboratory
81	Technical Support Department
85	Computer Department

APPROVED BY: P. M. Sturm DATE: 30 September 1977

P. M. STURM
Commander, USN
Deputy Director, AVID

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE. THE COPY FURNISHED TO DDC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NADC-77024-30	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A COMPUTERIZED VSTOL/SMALL PLATFORM LANDING DYNAMICS INVESTIGATION MODEL		5. TYPE OF REPORT & PERIOD COVERED FINAL REPORT 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Ronald L. Nave		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Aircraft & Crew Systems Technology Directorate Naval Air Development Center Code 6053 Warminster, PA 18974		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS AIRTASK WF41-421-203 Work Unit MX2X1
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Air Systems Command Department of the Navy Washington, D.C. 20361		12. REPORT DATE 30 September 1977 13. NUMBER OF PAGES 285
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) AV8A Harrier Simulation VSTOL/Ship Interface Dynamic Model		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) A digital computer program has been developed to investigate the dynamic interaction between VSTOL aircraft and ships. The model includes the ship, aircraft, pilot, and air turbulence. At present the aircraft model is configured to represent the AV8A Harrier, but other aircraft may be simulated by simple program changes.		

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

SUMMARY

A batch process digital computer program has been developed which can simulate the take-off and landing of VSTOL aircraft aboard ship. The program consists of a group of basic subroutines applicable to any aircraft and a set of specific aircraft model subroutines which have been configured to represent the AV8A Harrier. The aircraft model includes nonlinear aerodynamics, engine and reaction control response, stability augmentation and actuator dynamics, and a simplified landing gear model.

Ship dynamics are modeled as six degree of freedom sinusoidal motion. The ship is assumed to have a fixed mean position about which it oscillates. Wind over the deck is composed of a steady induced wind equal to the ship speed plus a separate North and East component of natural wind which can be independently specified. At present no turbulence model specifically designed for VSTOL aircraft exists, but a model developed for conventional carriers is incorporated. This subroutine calculates free air turbulence as well as ship wake turbulence which may be varied in amplitude. The wake intensity is calculated as a function of range, altitude, and lateral position relative to the flight deck.

A pilot model subroutine provides both open loop and closed loop control inputs. Thus far, the pilot model has been used only to investigate aircraft response, and, hence, it has not been constructed to accurately represent true pilot dynamics. The pilot model can now execute a short take-off and climb to specified altitude, and a side step from hover to a vertical landing using closed loop control as well as step, doublet, pulse, and sinusoidal inputs to each control.

Three data output options are provided. These consist of a printout of specified variables versus time, a statistical summary, and a Calcomp plot of a maximum of 14 variables versus time.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF FIGURES (Continued)

LIST OF TABLES

LIST OF TABLES (Continued)

LIST OF TABLES (Continued)

LIST OF TABLES (Continued)

INTRODUCTION

The anticipated deployment of VSTOL aircraft aboard small escort vessels necessitates the development of methods for analytically predicting the performance and handling characteristics of these aircraft in this relatively undefined environment. A thorough analysis of landing and take-off dynamics should include all of the components involved both as individual elements and as a total interactive system.

As a minimum, the following elements should be considered:

- Aircraft aerodynamics
- Engine and reaction control dynamics
- Stability augmentation and control system
- Landing gear, catapult, and arresting gear, if used
- Ship motion
- Atmospheric turbulence
- Ship induced turbulence
- Aircraft/Engine airflow interaction with the ship
- Pilot dynamics
- Pilot visual displays and landing aids
- Landing signal officer influence on pilot performance.

The method of approaching this problem was to develop a digital simulation program which incorporated as many of the system elements as possible in separate subroutines which could be changed easily. In addition, most system parameters have been placed in common arrays so that these values may be changed without recompiling the subroutines.

The program is divided into a group of basic routines which are independent of aircraft configuration. These basic routines handle trim initialization, coordinate transformations, the integration of differential equations, and the interpolation of tabulated nonlinear functions. Control inputs, forces and moments, and disturbance inputs are determined by user supplied routines which must be varied to represent specific aircraft types and atmospheric conditions.

Each of the VSTOL system elements is complicated and important enough to justify a large effort to investigate its impact. However, due to the limits of time, a decision was made to develop a simple description of all of the elements rather than concentrating the effort on one or two areas. This made it possible to develop an overall frame work to which more sophisticated and accurate model formulations may be added as they become available. The basic subroutines were adapted from a simulation program used at the NASA Ames Research Center to perform moving base real time simulations. The user routines were configured to represent the AV8A Harrier using the best data currently available.

At the present time, the model addresses the first nine of the eleven model components described above. No model description of the LSO or the pilot landing aids is presently incorporated because it would be very difficult to develop meaningful models of these two components. A piloted aircraft simulation with a high resolution wide field visual display is probably the only way other than flight testing to thoroughly evaluate

the LSO task and the effectiveness of various pilot displays and landing aids. A dual simulation including both the pilot and the LSO would probably be the ideal approach.

The aircraft model includes an aerodynamics subroutine which determines the three force and three moment coefficients by interpolating tabulated values of aerodynamic functions. A separate subroutine is used to represent the Pegasus 11 engine operating at low altitude and Mach number. Engine RPM, nozzle angle, and reaction control thrust dynamics are included in the equations.

A simplified model of the landing gear including vertical force, braking force, and pitching moment is also incorporated.

GENERAL PROGRAM DESCRIPTION

The VSTOL simulation program currently consists of a main program, 31 subroutines, and a data input deck. These subroutines are functionally divided into four groups as follows:

Trim Routines
- Setup
- Bsetup
- Bquiet
- Minv

Basic Equation Calculations
- Loop2
- Loop3
- Bveloc
- Btrans
- Brotate
- Balfbet
- Batmos
- Lookup
- TAB1
- ARDC62
- RAND

User Supplied Model Structure
- Pilot
- Engine
- Aero2
- ContR2
- Gears
- WINDC
- Ship
- Deck

Data Input/Output
- DatSav
- DPlot
- Timehis

Show
Printo
Tabrd
BICPRI
Block Data

System routines are used to generate a CALCOMP plotting tape, and to invert the state matrix during the trim sequence.

S U B R O U T I N E D E S C R I P T I O N

MAIN PROGRAM

The main program controls input and output of data and the calling sequence of the basic and user subroutines. A series of runs may be made to investigate various configurations with changes between each run. Also, multiple runs with a fixed configuration may be made to evaluate the statistical response to turbulence or other external disturbances.

The main program loads all function tables by calling subroutine TABRD. This subroutine loads all nonlinear function tables required by the simulation from data cards contained in the load deck each time the program is run. These tables include aerodynamics, thrust functions, control gearing, landing gear forces, and turbulence functions.

After all parameters are specified for a given run, a trim calculation is performed which adjusts specified control parameters until all forces and moments are balanced. This is achieved by cycling through subroutine setup. When the trim criteria is satisfied, the switch parameter ITPROG is set to 0. This causes the trim process to terminate and calls subroutine BICPRI which prints the initial values of all aircraft states and control positions. The program currently performs all calculations in English units and converts them to metric units for the trim printout. Inputs are specified in English units.

The dynamic portion of the program next calculates the aircraft time response to specified control inputs by cycling through the two major subroutines LOOP2 and LOOP3. The run is terminated when the running variable time exceeds the specified maximum routine variable RTIME. If a landing approach is simulated, the run will be terminated when the landing gears touch the deck or runway provided parameter JLAND is set to 1 before the run and ISTAT = 1.

Data output is provided by selecting values for parameters ITHIS, IUPLOT, and ISTAT. ITHIS = 1 provides a time history printout. IUPLOT = 1 causes a CALCOMP plot tape to be loaded, and ISTAT = 1 provides a statistical summary of each run as well as consecutive run landing statistics.

Print variables may be changed from run to run by specifying IREAD1 = 1. Plot variables are changed by setting IREAD2 = 1. These flags

cause new values of arrays IPRINT(I) and IPLOT(I) to be read. Each number in these integer arrays specifies one variable to be plotted or printed as well as selecting either English or metric units for the output.

PILOT

Subroutine PILOT is designed to provide control inputs representative of pilot action. A variety of control strategies are currently included which may be selected by choosing values of variables ITASK, NCHK, and MCHK. The current control laws were established with the intention of achieving a stable rapid response rather than accurately simulating true pilot capabilities and dynamics. The following control strategies are currently programmed:

ITASK = 1	Release brake, begin takeoff roll, retard throttle and brake to a stop.
ITASK = 2	Execute a rolling vertical takeoff, transition to wing borne flight, climb to a specified altitude, level off, and hold altitude.
ITASK = 3	Execute a side step and vertical landing starting from a hover relative to the landing point.
ITASK = 4	Move the throttle sinusoidally about a hover trim condition.
ITASK = 5	Open loop control inputs as specified by variables NCHK and MCHK.

MCHK = 1	Throttle input
MCHK = 2	Longitudinal stick input
MCHK = 3	Lateral stick input
MCHK = 4	Nozzle angle input
MCHK = 5	Rudder input

NCHK = 0	No input
NCHK = 1	Step input
NCHK = 2	Pulse input
NCHK = 3	Doublet input
NCHK = 4	Sinusoidal input

Subroutine PILOT provides lateral and longitudinal stick inputs, rudder pedal, throttle, nozzle commands, plus discrete brake application and water injection commands. In addition to the described options, the longitudinal stick may be matched to flight measured values of elevator position. If parameter I2DOF = 1, the stick position is calculated by first subtracting the calculated value of elevator due to pitch SAS from the tabulated value of total elevator versus time. The stick position is then determined by interpolating tabulated values of stick versus elevator position.

BLOCK DATA

Block data is used to initialize parameters located in common arrays.

CONTR2

CONTR2 calculates aileron, elevator, and rudder position commands based on pilot inputs. It also simulates the operation of the stability augmentation system (SAS) and the elevator and aileron power actuators. The pitch SAS is driven by a pitch rate signal modified by a gain and a lead-lag network. This command drives the stabilizer and the rear pitch jet. The SAS operation is nonlinear at low speeds because the front pitch jet is not connected to the stabilization system, and the SAS cannot generate nose up moments in hover. Roll damping is improved by driving the ailerons and roll reaction jets with a filtered roll rate signal. The lateral dynamics are augmented by driving the yaw reaction jet with yaw rate and lateral acceleration. All three SAS channels include position command limits. In addition to direct control links, there is an interconnect from the aileron to the yaw reaction jet which counteracts adverse yaw characteristics. The three channels of the stability augmentation system may be individually disconnected by setting variables IQDAMP, IPDAMP, or IRDAMP to zero. These correspond to the pitch, roll, and yaw dampers, respectively. Figure 1 illustrates the SAS.

A special 2 degree of freedom mode may be selected for use in comparing flight data with simulated data. When parameter I2DOF is set to 1, the elevator follows a tabulated function of time rather than the value dynamically computed by the computer.

GEARS

Subroutine GEARS provides a simplified model of the landing gear longitudinal forces and the pitching moment. The model assumes that all gear forces are produced by the nose and main gears. The compression of each landing gear element is calculated as being equal to the difference between the undeflected gear height and the height of the deck directly beneath each wheel. Force is assumed to act normal to the deck with a magnitude determined as a tabulated function of compression. An additional force term is added which is proportional to the rate of compression of each strut. This term was sized to provide a reasonably well damped response, but is not based on measured data. Pitching moment is calculated by multiplying the nose and main gear normal force and friction force by their respective moment arms.

Axial force is assumed to be proportional to the normal force and acts in a direction opposite to the relative velocity or tendency toward motion between the deck and the aircraft. Based on experimental data, the rolling friction coefficient is assigned a value of .03 for free rolling and .6 with brakes applied. All forces are generated in deck axes and are then transformed into body axes. Body axis Z force is limited to acting in an upward or negative Z direction. The gear model is illustrated in Figure 2.

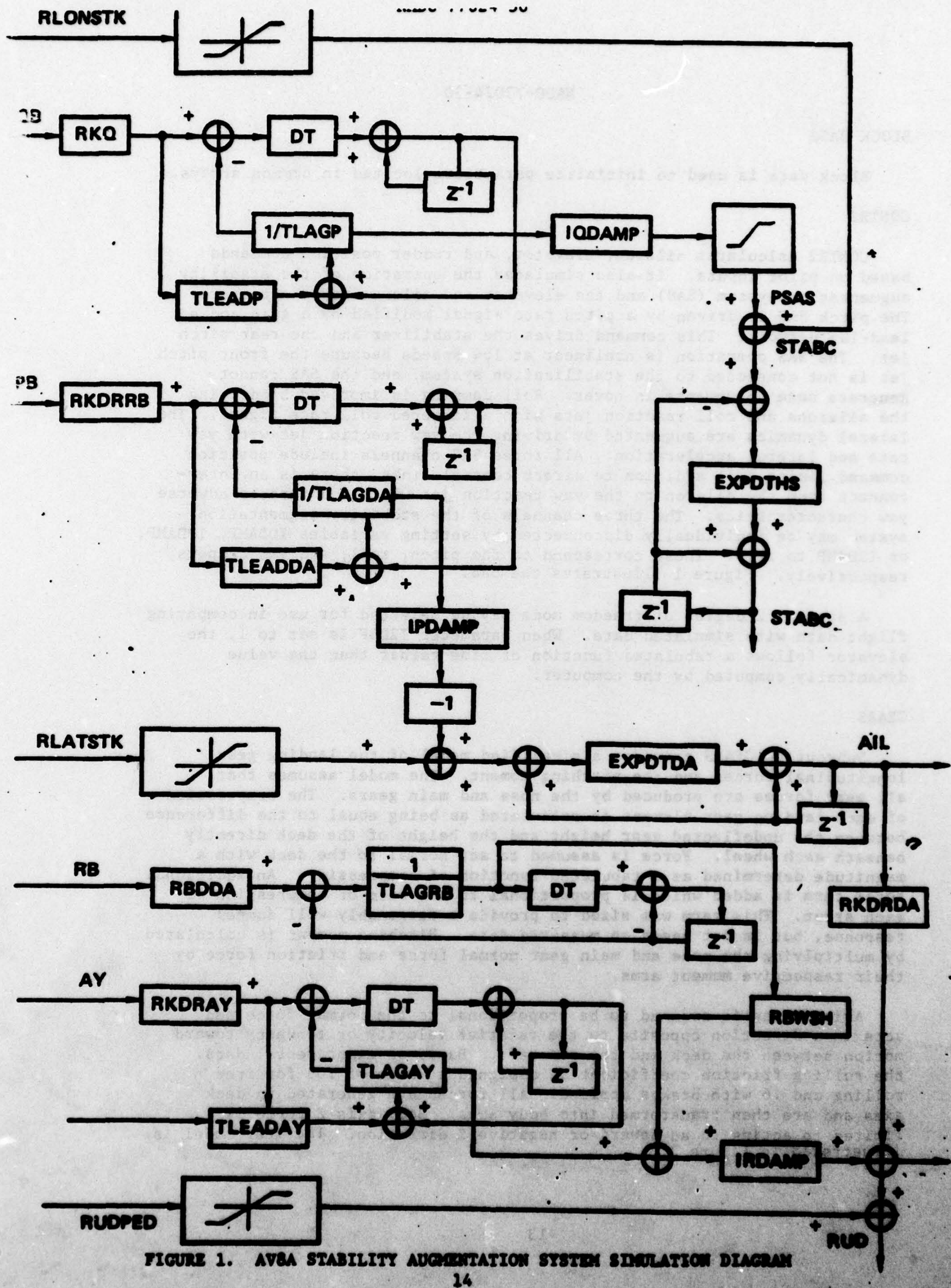


FIGURE 1. AV8A STABILITY AUGMENTATION SYSTEM SIMULATION DIAGRAM

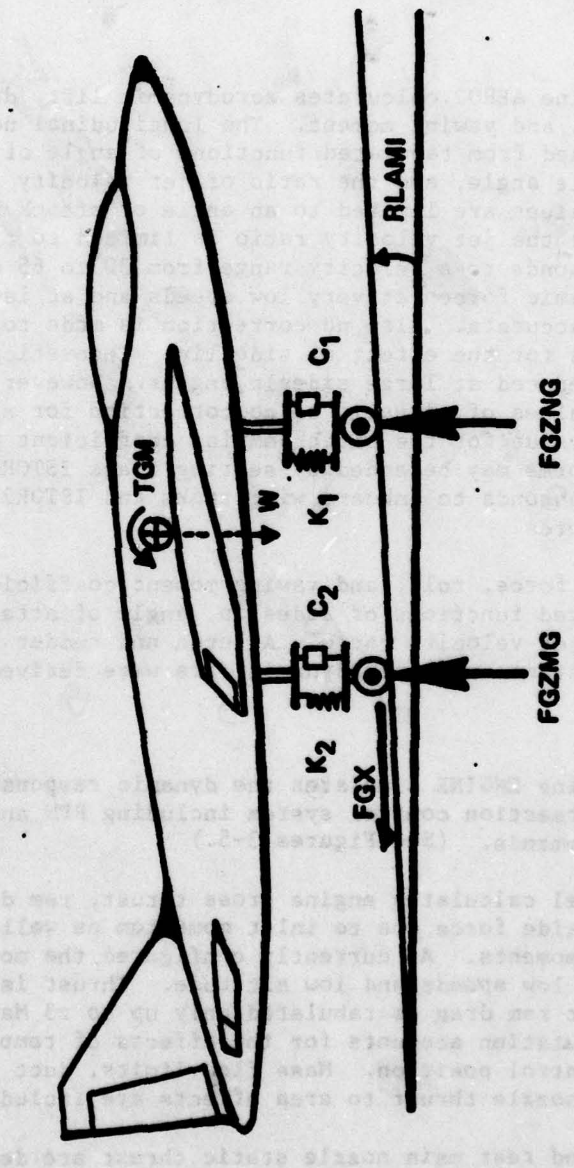


FIGURE 2. LANDING GEAR FORCE DIAGRAM

No roll or yaw moment or side force is currently included in the landing gear equations. However, a clamp is placed on the roll angle and roll rate when both the left and right wing gears are in contact with the deck. The aircraft bank angle is set equal to the deck tilt angle as measured along the aircraft Y axis. An additional clamping function is added to attitude and pitch rate when the main and nose gear are fully compressed. Finally, a flag is set when the gear deflection exceeds the maximum value. This flag may be used to terminate the program. This is indicated in the program by the statement ICRASH = 1.

AERO2

Subroutine AERO2 calculates aerodynamic lift, drag, side force, roll, pitch, and yawing moment. The longitudinal non-dimensional coefficients are determined from tabulated functions of angle of attack, stabilizer, engine nozzle angle, and the ratio of jet velocity to airspeed. The tabulated values are limited to an angle of attack range from -5 to +20 degrees, and the jet velocity ratio is limited to a range from 5.4 to 11.9. This corresponds to a velocity range from 30 to 65 m/sec. As a result, the aerodynamic forces at very low speeds and at large angles of attack may not be accurate. Also, no correction is made to the longitudinal coefficients for the effect of sideslip. Theoretically, the wing lift should be reduced at large sideslp angles. However, no data is available for large values of sideslip so no correction for sideslip is included. A constant value for the pitch damping coefficient was assumed. Corrections for wing stores may be added by setting flags ISTOR1 and ISTOR2 = 1. ISTOR1 corresponds to inboard wing tanks and ISTOR2 corresponds to outboard stores.

Lateral force, roll, and yawing moment coefficients are calculated from tabulated functions of sideslip, angle of attack, engine nozzle angle, and jet velocity ratio. Aileron and rudder effectiveness are assumed constant. All aerodynamic data were derived from reference (a).

ENGINE

Subroutine ENGINE simulates the dynamic response of the Pegasus 11 engine and reaction control system including RPM and nozzle angle response to pilot commands. (See Figures 3-5.)

The model calculates engine gross thrust, ram drag, pitch and yaw moment and side force due to inlet momentum as well as reaction control forces and moments. As currently configured, the model is intended for use only at low speeds and low altitude. Thrust is varied with air density, but ram drag is tabulated only up to .3 Mach. The reaction jet thrust computation accounts for the effects of temperature, air density, RPM, and control position. Mass flow limits, duct losses, and non-linear RCS nozzle thrust to area effects are included.

Front and rear main nozzle static thrust are determined from tabulated functions of RPM and reaction control mass flow. Total gross thrust is determined by adding an increment to the static thrust which depends on RPM and Mach number. Ram drag is calculated as a function of RPM and

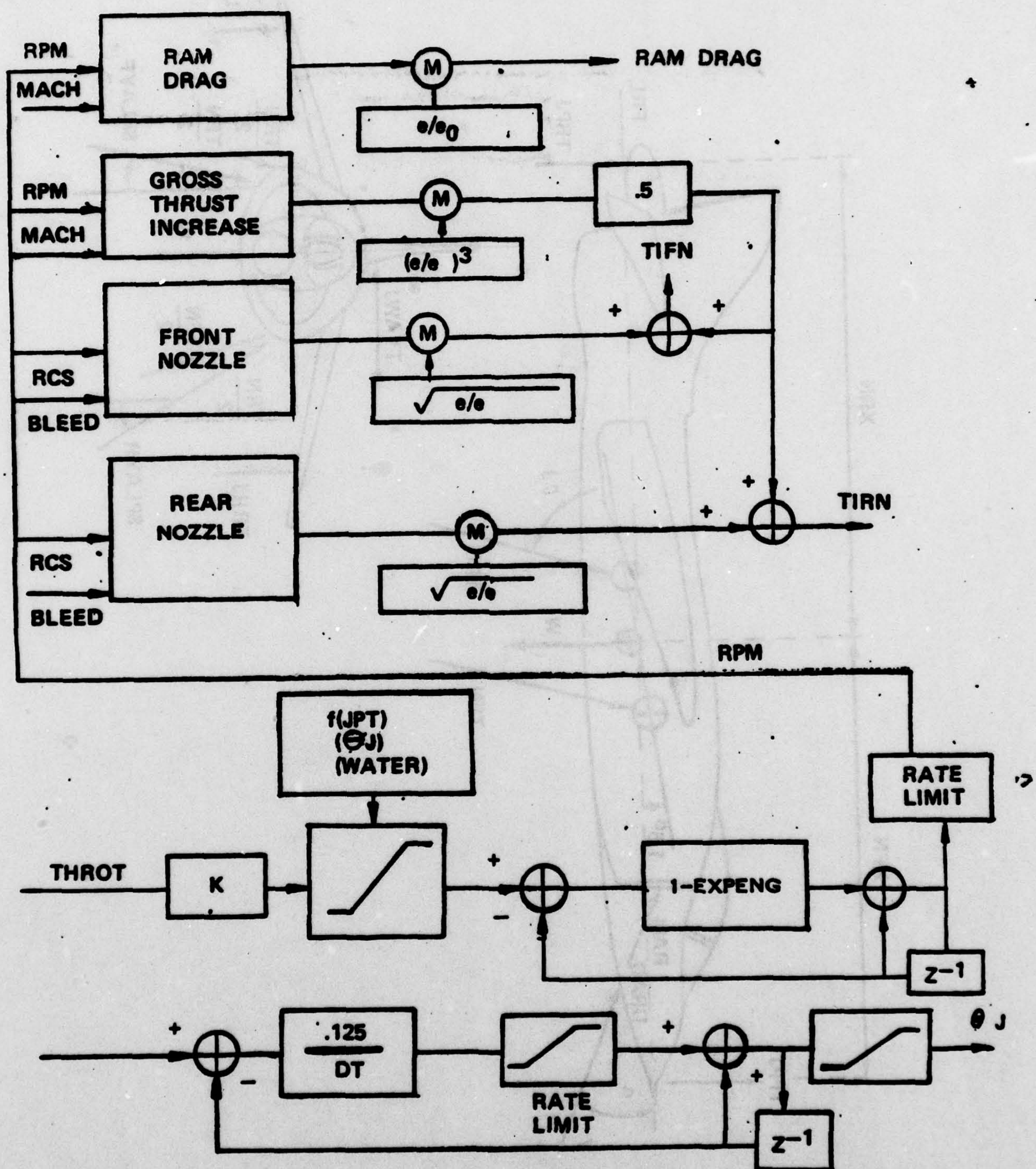


FIGURE 3. ENGINE SIMULATION DIAGRAM

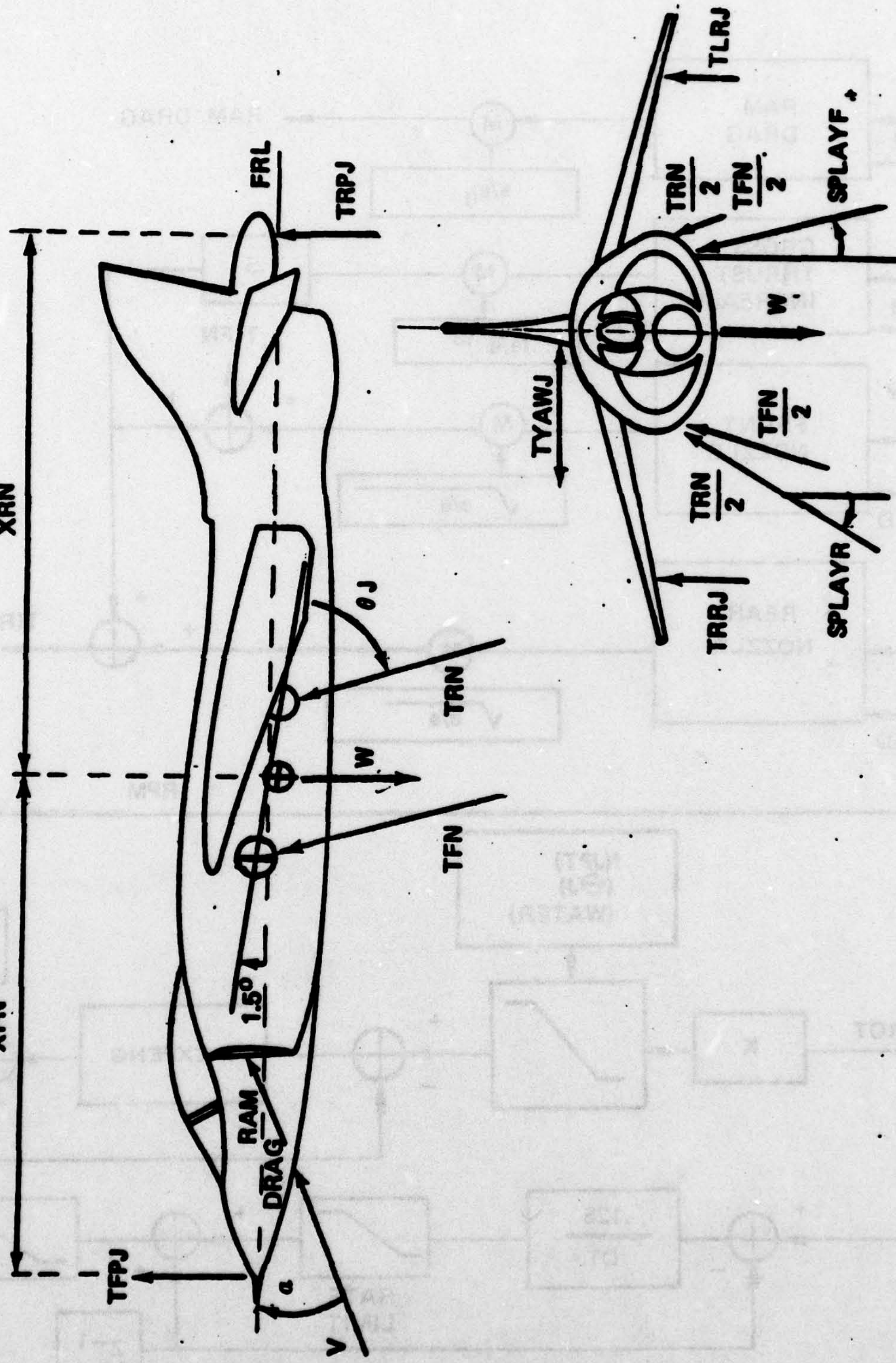


FIGURE 4. ENGINE FORCE DIAGRAM

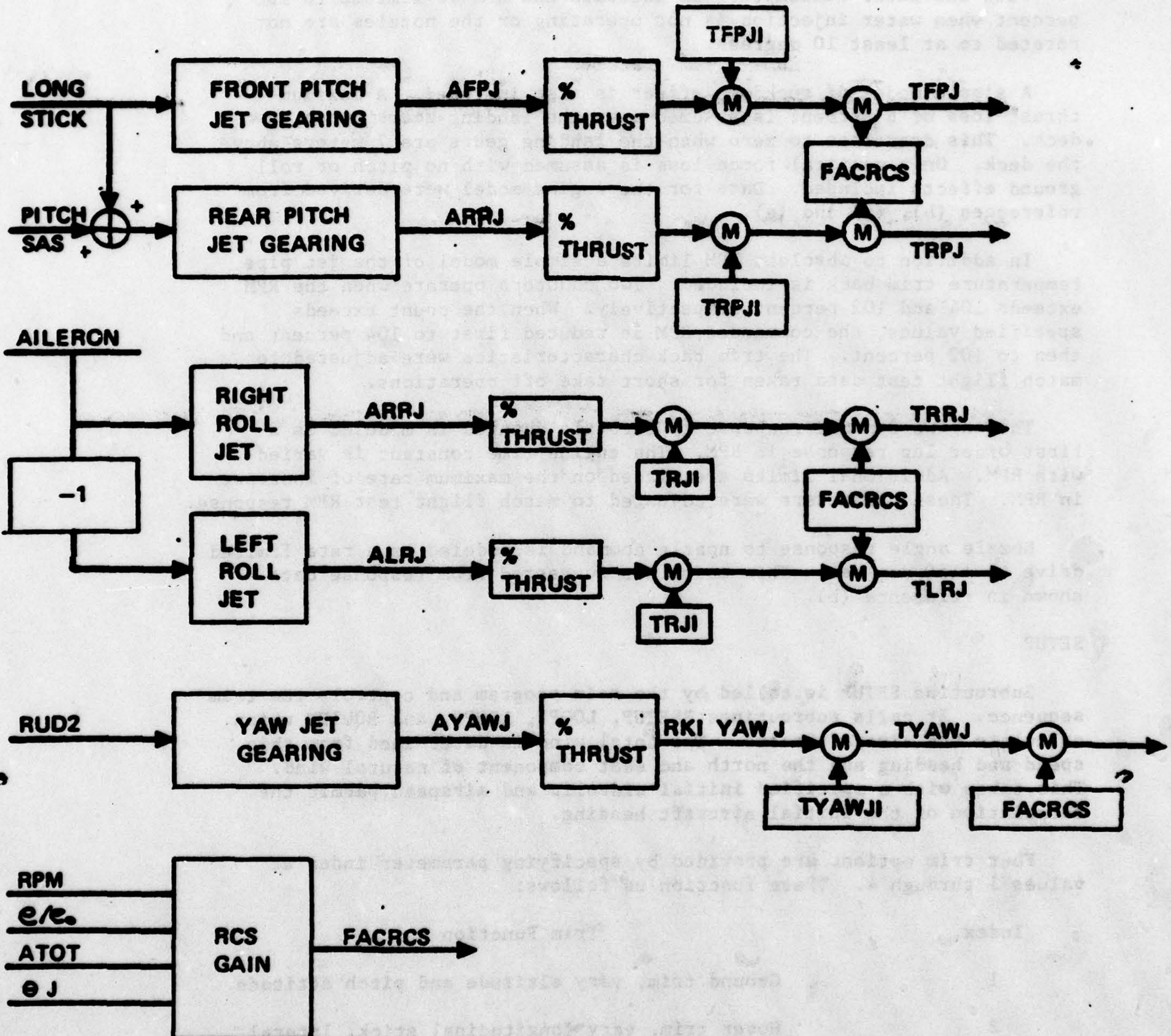


FIGURE 5. REACTION CONTROL SYSTEM SIMULATION DIAGRAM

Mach number. In all cases, engine characteristics at low RPM were extrapolated from data available at an RPM of 85 percent or higher. Lateral and longitudinal orientation of each lift and control nozzle is accounted for and the total forces and moments are summed in body axes.

Fuel and water consumption is included and RPM is limited to 102 percent when water injection is not operating or the nozzles are not rotated to at least 10 degrees.

A simple model of suckdown effect is also included. A maximum thrust loss of 6 percent is assumed when the landing gears touch the deck. This decreases to zero when the landing gears are 7 meters above the deck. Only vertical force loss is assumed with no pitch or roll ground effects included. Data for the engine model were derived from references (b), (c) and (e).

In addition to absolute RPM limits a simple model of the jet pipe temperature trim back is included. Two counters operate when the RPM exceeds 104 and 102 percent, respectively. When the count exceeds specified values, the commanded RPM is reduced first to 104 percent and then to 102 percent. The trim back characteristics were adjusted to match flight test data taken for short take off operations.

The engine dynamic response to throttle changes is modeled as a first order lag response in RPM. The engine time constant is varied with RPM. Additional limits are placed on the maximum rate of increase in RPM. These parameters were adjusted to match flight test RPM response.

Nozzle angle response to nozzle command is modeled as a rate limited drive of ±150 deg/sec. This model was suggested from response data shown in reference (b).

SETUP

Subroutine SETUP is called by the main program and controls the trim sequence. It calls subroutines BSETUP, LOOP2, LOOP3, and BQUIET which calculate the aircraft state. The total wind is determined from ship speed and heading and the north and east component of natural wind. This, taken with a specified initial sideslip and airspeed, permit the calculation of the initial aircraft heading.

Four trim options are provided by specifying parameter index at values 1 through 4. These function as follows:

Index	Trim Function
1	Ground trim, vary altitude and pitch attitude.
2	Hover trim, vary longitudinal stick, lateral stick, throttle, pitch attitude, rudder, and bank angle.

3	Transition flight - vary longitudinal stick, throttle, lateral stick, nozzle angle, rudder, and bank angle.
4	Conventional flight - vary longitudinal stick, airspeed, lateral stick, throttle, rudder, and sideslip.

DATSAV

Subroutine DATSAV loads an array with values of 14 variables in either English or metric units plus the value of time in seconds. The 14 variables and the units of the variable are determined by integer array IPLOT. Each value of IPLOT specifies the conversion factor, the array, and the array element to be plotted. The 3 right digits of the value of IPLOT(I) specifies the array element to the plotted. These range from 1 to 200 for the B array and 1 to 500 for the A array. The fourth digit from the right specifies the array with 1 = A and 2 = B. The left two digits specify the conversion factor required to convert from English units to plotting units which may be either English or metric. The last two digits may range from 01 to 12 as follows:

Digit Value	Conversion Factor	
01	1	nondimensional or English print
02	4.4418	LBF to Newton
03	.45359	LBM to kilogram
04	14.5919	Slugs to kilograms
05	.3048	Feet to meters
06	.092903	$Feet^2$ to $meter^2$
07	.0283168	$Feet^3$ to $meter^3$
08	515.3089	$Slug/feet^3$ to $kilogram/meter^3$
09	47.8115	$Pound/feet^2$ to $newton/meter^2$
10	1.3538	$Slug\text{-}foot^2$ to $kilogram\text{-}meter^2$
11	6.4516 $inch^2$	to $centimeter^2$
12	2.54	inch to centimeters

In addition to storing the plot variables in appropriate dimensional form, DATSAV also determines the maximum and minimum value of each plot variable. These values are stored in the array PSCALE which may be used to automatically scale the plots.

PRINTO

Subroutine PRINTO provides a statistical summary giving minimum, maximum, RMS, and average values of selected variables specified in subroutine *SHOW*. *In addition touchdown and deck edge crossing points* are recorded, and a statistical summary of the touchdown data is provided for multiple runs.

SHOW

Subroutine SHOW,which is called by LOOP2 when ISTAT = 1,collects data on 17 parameters selected by the user. . The maximum, minimum, average, and RMS value of each parameter is calculated at each program iteration.

Data gathering is divided into 3 segments depending on airspeed and nozzle angle. Airspeeds less than 30 meter/second with nozzle angles greater than 60 degrees and wheels off the ground designate a hover condition. Airspeeds greater than 30 meter/second with nozzle angles greater than 20 degrees and wheels off the ground designate a transition condition. Airspeed greater than 30 meter/second with nozzle angles less than 20 degrees designate conventional flight. In addition to the data averaging, discrete data points are stored when the aircraft crosses the deck edge and when the wheels touch the deck.

DPLOT

Subroutine DPLOT loads a Calcomp magnetic tape which will produce plots having a format identical to that of an 8 channel brush recorder plot. This permits the direct overlay of simulation plots and flight test plots. The variable magnitude is plotted along the Y axis, and time is plotted along the X axis. Plot time scale is specified by variable CHTSP. The ordinate scale is specified by the array scale which determines the maximum and minimum value of each of the 14 plots. The overall size of the plots is proportional to the value of variable FAC. The standard value is 1.0 for matching brush records. If variable ISCALE = 1, the present scale factors are ignored and the plots are scaled on the basis of the maximum and minimum values encountered during the run. Plot abscissa titles are varied automatically when the plot variables are selected. The plot titles are loaded in block data for all variables stored in the A and B areas.

TIMEHIS

Subroutine TIMEHIS,which is called by AV8A2,prints 11 selected variables plus the value of time at specified time intervals. The time interval is determined in the main program by variable DTIM2. The variables and the choice of English or metric units are specified by array IPRNT(11) in the same way that the plot variables are specified by IPLOT(I). Both IPLOT(14) and IPRNT(11) may be updated from run to run as specified in the main program.

WINDC

WINDC provides a model of free air turbulence and ship air wake. The model was developed for conventional carrier landing analysis. As such,it is not specifically representative of the air wake environment around a small platform ship. However, the inputs provide a qualitative indication of gust response. It is planned to modify this subroutine as soon as wind tunnel results of ship model tests become available. Random turbulence is simulated by filtering the output of a Gaussian

random number generator in order to shape the frequency content of the simulated turbulence. Five distinct random turbulence components are generated. Three represent the orthogonal components of free air turbulence which are assumed to have constant statistical properties. Two components are used to represent the vertical and horizontal random components of the ship wake. The random burble varies in magnitude and frequency with range to the ship.

In addition to the linear filtering network, the random number sequences are multiplied by a unit magnitude sine wave. This has the effect of shifting the center frequency of the random number sequence by an amount $\pm W$ where W is the frequency of the sine wave. The value of W may be selected to match simulated turbulence with measured turbulence spectra.

The linear filter used to shape the signals cuts off the high frequency components of the random number generator spectrum. Filter coefficients are calculated as functions of trim airspeed and integration step size. An increase in airspeed results in an increase in the frequency of the turbulence. Because of the formulation, the turbulence magnitude approaches zero as the airspeed is reduced. Therefore, a minimum value of 20 meters per second is specified in order to provide disturbances at low airspeed. This is a somewhat arbitrary fix, and its accuracy is unverified. The minimum value was selected as a representative wind overdeck.

The U and W components of the free air turbulence are each processed through a first order filter. The V gust component is filtered through a second order filter. These filters are intended to duplicate the gust spectrum specified in MIL SPEC AR-40. Although the free air turbulence is calculated in body axis components, it is assumed to be independent of aircraft position and orientation.

The ship wake portion of the turbulence, however, is a function of ship speed, ship pitch, aircraft position relative to the ship, and aircraft attitude. The wake intensity is somewhat arbitrarily assumed to decrease exponentially as the aircraft moves left or right of the extended ship center line. No provision is made for the expanded wake that occurs when the ship is yawed with respect to the relative wind. In addition to lateral limits on the wake, the wake is assumed to vanish above a specified altitude which increases with range behind the ship. Within the altitude and lateral bounds, the steady and random components of the wake vary with range. The burble is composed of seven components. They are as follows:

UBSS	Steady U velocity degradation
WBSS	Steady W velocity upwash/downwash
UB1	U velocity due to deck pitching velocity
WB1	W velocity due to deck pitching velocity
UB2	Random U burble velocity

WB2	Random W burble velocity
VYBR	Random lateral burble velocity

UBSS and WBSS are functions of range and wind over the deck. The deck motion induced wake components UB1 and WB1 are functions of closing speed, ship pitch and range behind the stern. The random components of the burble UB2 and WB2 are functions of range to touch down and ship speed. VYBR, the lateral component of the burble, is assumed equal to the vertical component WB2. Burble components are computed in ship axes, first converted to north, east, down inertial axes, and are then transformed into aircraft body axes. The three burble components are then summed with the free air turbulence components to form a total turbulence velocity vector.

Various turbulence options may be selected by setting appropriate switches. Any one of ten random sequences may be repeated by setting variable IONCE equal to values from one to ten. If IONCE = 0, the random sequence will continue from one run to the next. In this way, various airplane configurations may be subjected to identical turbulence conditions, or a particular configuration may be repeated and subjected to non-repetitive turbulence in order to determine the statistical characteristics of the response of a fixed aircraft/pilot configuration. The burble characteristics may be held constant by setting IBFRZ = 1. This causes the burble model to act as though the aircraft was at a fixed range XFRZ behind the ship. Setting IWIND = 0 eliminates all turbulence while ILTURB = 0 sets the free air turbulence components equal to zero. Variables SUBSW and SWBSW provide magnitude scale factors on the steady burble components. Finally, the magnitude of the random components are selected by setting values for the seven elements of the array GR(I).

In addition to calculating the current values of the turbulence vector, WINDC also calculates the statistics of each wind component including average and RMS values.

SHIP

Subroutine SHIP provides a six degree of freedom representation of ship motion. The equations are adapted directly from reference (d) and represent the motion of a DLG-26 class destroyer in sea state 5 with 10 knots ship speed, 13 meter/sec (26 knots) wind and 150 degree heading with respect to the wave crests. This approximates a worst possible condition in which flight operations may be attempted.

SUBROUTINE DECK

Subroutine DECK computes the inertial position of the ideal touchdown point and the velocity and height of the deck at a point directly beneath each of the four landing gears. It sets a flag IOVDK to one if the aircraft is over the deck edge, and sets array element ITOUCH(I) = 1 when the Ith element of the four landing gears touch the deck. This information is used to calculate landing gear forces and to determine when touchdown occurs.

LOOP3

LOOP3 is part of the basic simulation package developed by NASA Ames. It integrates aircraft inertial velocity to determine position. It also calculates vertical and horizontal flight path angles. LOOP3 is called by the main program at a normal rate of 10 times per second. It in turn calls subroutine DECK which computes ship motion and landing gear position.

LOOP2

LOOP2, which is called by the main program at a normal rate of 20 times per second, computes body axis forces and moments and integrates the forces to determine inertial velocities and altitude. It calls brotate which integrates the torque equations, and BTRANS which calculates the Euler angle transformations. BVELOC, BALFBET, SHOW, PILOT, CONTR2, ENGINE, AERO2, and WINDC are also called by LOOP2.

In addition to the basic function, a landing gear position clamp is included. When the brakes are set, the wheels are in contact with the deck, and the relative velocity between the wheels and deck is less than two feet per second; the aircraft position is fixed relative to the deck. This allows the aircraft to brake to a stop without oscillating back and forth as would occur without the clamp.

BQUIET

Subroutine BQUIET computes a gradient matrix relating aircraft state error u, v, w, Q, P, R to control inputs. The error matrix is used to compute a weighted least square error matrix which is non-singular even if the number of controls is less than the number of states. The error squared matrix is inverted by subroutine MINV and used to compute a gradient step of the selected control variables intended to minimize the state error. Slight modifications were made to the NASA Ames version of this subroutine which appear to improve the efficiency somewhat. In contrast to the original version, the gradient is evaluated only once at the beginning of the trim procedure. Then a maximum of 25 steps are taken with the fixed gradient. After each step is taken, the trim error is evaluated. If the weighted squared error is less than 90 percent of the past value of this error, the old gradient is retained. Otherwise, a new gradient is calculated. Also, the step size is reduced after each step to ensure convergence. When the gradient is reevaluated, the step gain is returned to the initial value. A special trim sequence is provided for trimming the aircraft on the deck in which only altitude and pitch attitude are varied.

The trim sequence is terminated when the error criterion is satisfied or the maximum trim iteration count is exceeded. A failure to trim within the specified number of iterations usually indicates some error in the aircraft model structure. Providing a good initial guess for the trim variables will greatly improve the rate of convergence.

BVELOC

Subroutine BVELOC computes body axis turbulence given inertial turbulence or inertial turbulence components given body axis components depending on the value of IETURB. It also computes the inertial wind, aircraft relative inertial velocity, and body axis relative wind components.

BTRANS

BTRANS computes the sine and cosine of the aircraft elevation, bank, and heading angles. These six values are then used to compute the nine elements of the Euler angle transformation matrix. This matrix is used to transform position, velocity, and acceleration from body axes to inertial axes. The reverse transformation is achieved using the transpose of the matrix.

BROTATE

Subroutine BROTATE computes the rotational motions of the aircraft. Angular accelerations are computed in body axes and integrated to yield pitch, roll, and yaw rates. These rates are converted to inertial Euler angle rates which are then integrated to yield elevation, bank, and heading angles. In addition to the basic function, a heading clamp is provided. This serves as a substitute for landing gear yawing moment when the aircraft is on the deck. Changes in yaw angle are prevented when both the main and nose gear touch the deck.

The angular acceleration equations are formulated to include constant magnitude rotor angular momentum and cross products of inertia. Changes in rotor speed or configuration are not accounted for at present. However, the inertial coefficients could be computed and updated during the simulation run if they were required.

A more basic limitation occurs because the angular rate transformation is ill-defined at pitch angles of ± 90 degrees. Thus,the simulation cannot be used for spins or vertical climb or dive maneuvers.

A special reduced degree of freedom option is provided for use in matching flight test data. When the switch I2DOF is set to one, the computed pitch attitude and pitch rate are ignored and tabulated values of pitch attitude versus time obtained from flight measurement are substituted. The tabulated values are also differentiated to determine pitch rate. Using this procedure, the aerodynamic and engine parameters of the simulation may be more easily adjusted to match measured flight trajectories.

BALFBET

Subroutine BALFBET computes the aircraft angle of attack, sideslip, rate of change of sideslip and angle of attack and the airspeed given the three body axis components of velocity, and the body axis angular rates.

BATMOS

Subroutine BATMOS determines air density, the speed of sound, Mach number, dynamic pressure, total and ambient temperature, total and ambient pressure, and the equivalent airspeed based on standard atmospheric properties. It corrects for non-standard temperature based on a temperature increment specified by the user.

LOOKUP

Subroutine LOOKUP handles the linear interpolation of tabulated functions of 1, 2, or 3 variables. One of three search modes may be selected by setting variable meth equal to 1, 2, or 3. In addition, if the arguments exceed the tabulated values, option switches permit linear extrapolation, the return of the last value, or the setting of an error flag.

TAB1 (A, B, C, N, ND)

TAB1 returns to the calling program a single value determined as a function of the calling arguments. The first three parameters are arguments of the LOOKUP table, the fourth specifies the first table number referenced and the last parameter specifies the number of tables referenced. If the function depends on one or two parameters, only one table must be supplied for each value of the third variable. In addition to supplying the function value, TAB1 also prints an error message when the table arguments exceed the tabulated bounds.

TABRD

TABRD is called only during the setup procedure. It reads tabulated data points used in the interpolation routine from cards and loads them into designated arrays. At present,all of the tables must be loaded as a block when the deck is loaded. Individual tables cannot be updated between runs without reloading the program.

RAND

Subroutine RAND is a standard library subroutine which returns a random number each time it is called. Repeated calls to this subroutine produces a sequence of normally distributed numbers with zero mean and unit variance.

ARDC62

This subroutine contains the standard atmospheric density and speed of sound tables which are valid from 0 to 240,000 feet above sea level. Values of density and speed of sound are tabulated every 2000 feet. The subroutine returns one value of density and one value of the speed of sound given altitude as an input.

BSETUP

Subroutine BSETUP initializes aircraft position, velocity, and attitude during the trim procedure. It calculates initial values of weight and the inertial coefficients used by BROTATE to integrate the torque equations. Aircraft position can be specified with respect to the pilot eye location if ICG = 0, or with respect to aircraft cg if ICG = 1.

Angular velocities are initialized in body axis rates if IEULR = 0, and by Euler angle rates if IEULR = 1.

Airspeed may be initialized in knots, or Mach number, depending on the value of IMACH. It may be also specified by setting the values for the three components of body axis velocities, UB, VB, and WB. This is equivalent to specifying airspeed, angle of attack, and sideslip. For conventional flight, trim angle of attack is fixed and trim pitch attitude is varied to achieve the desired initial flight path angle.

MINV (HTH, M, DET, LM, MM)

Subroutine MINV is a standard library routine called during trim. It calculates the inverse of a matrix and the determinant. Array HTH initially contains the original matrix, and it holds the inverse after the subroutine returns. M specifies the dimension of the array, and DET is the value of the determinant. LM and MM are single dimensioned arrays equal in size to the dimension of the two dimension array.

PRELIMINARY SIMULATION RESULTS

A series of simulation runs have been made in an attempt to verify the accuracy of the computer model. To date, this attempt has been limited by the scarcity of fully instrumented open and closed loop aircraft response data. The only data available thus far has been from short take-offs made aboard ship. No data has been available for hover or conventional flight.

Although no comparison flight data was available, it was known that the Harrier is unstable although controllable in hover and low speed flight. Simulation results illustrated in Figures 6-11 agree with this assumption. With stability augmentation off, the aircraft diverges in all axes when subjected to an elevator doublet starting from a jet supported trim condition of 0 and 18 m/sec airspeed. If the stability augmentation system is engaged, the pitch response is stabilized, but the airspeed still appears to diverge. The simulation predicts an aperiodic divergence in roll with SAS off. With the SAS engaged, the lateral response is greatly reduced, but the aircraft still is unstable.

Figure 12 illustrates the aircraft pitch response at 51 m/sec with SAS off. This simulation predicts that the aircraft is unstable and has a very rapid aperiodic divergence. This agrees with reference (b) which predicts a time to double amplitude of 1.95 seconds at a 51 m/sec trim.

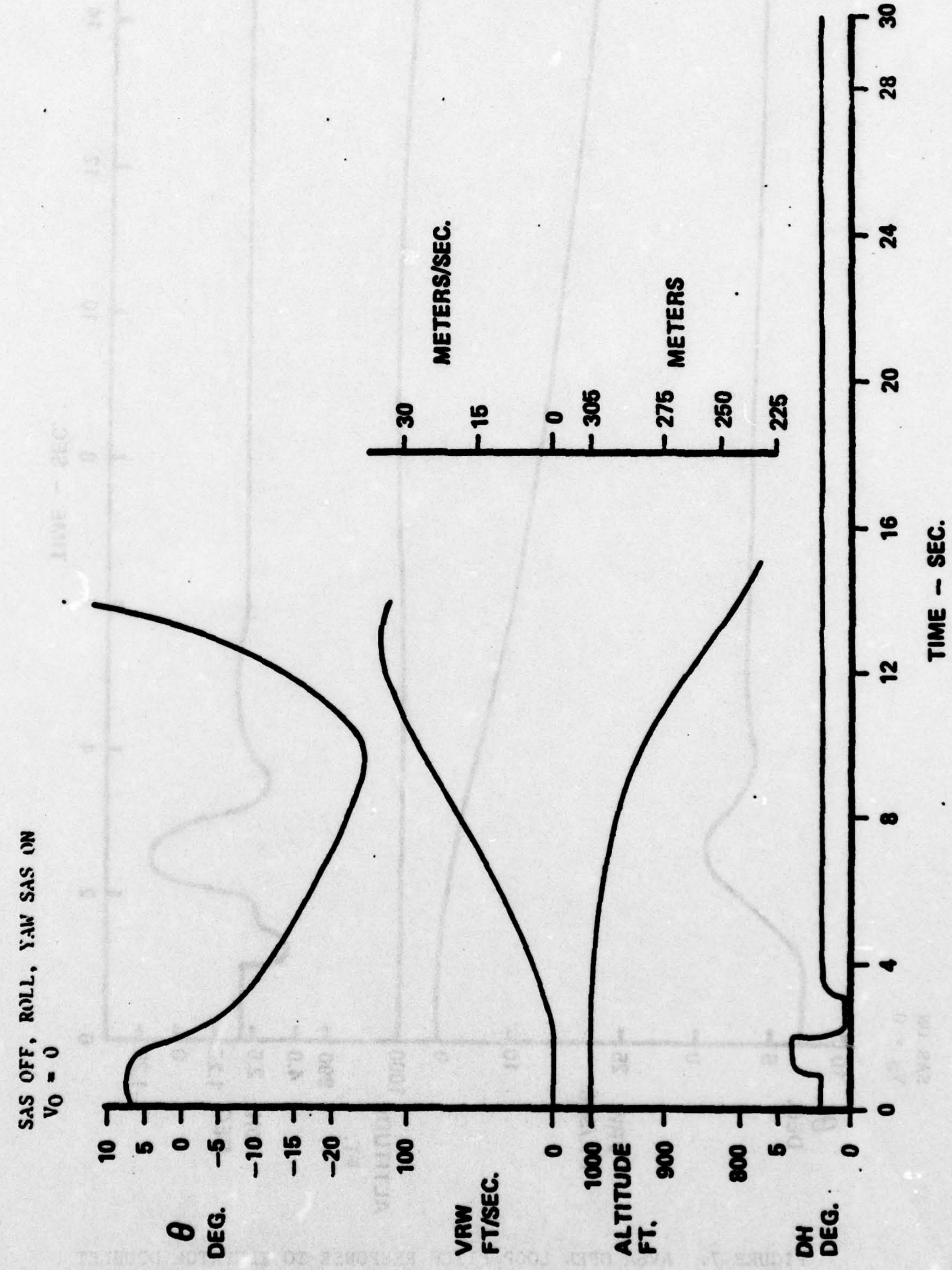


FIGURE 6. AV8A OPEN LOOP RESPONSE TO ELEVATOR DOUBLET

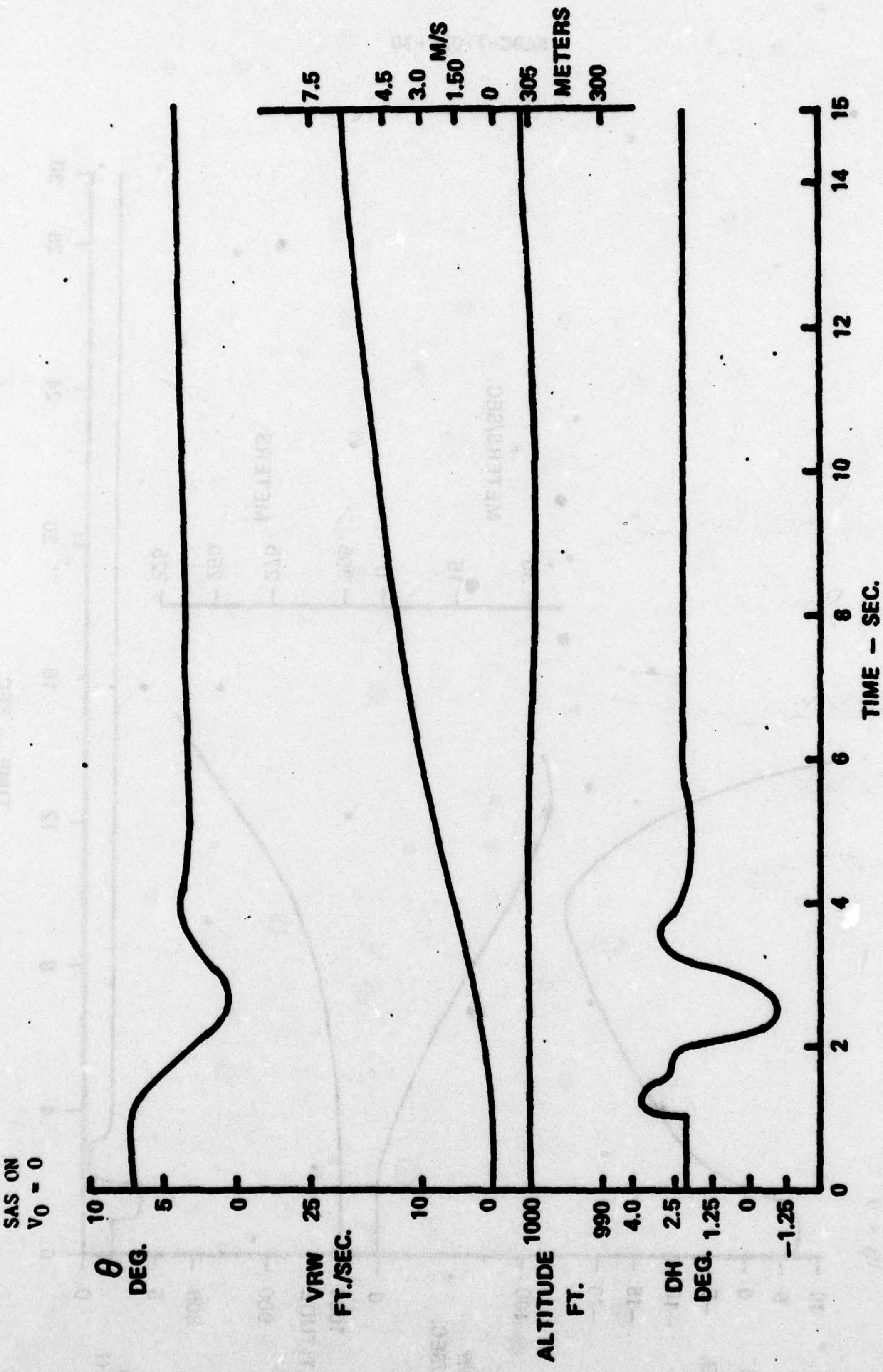


FIGURE 7. AV8A OPEN LOOP PITCH RESPONSE TO ELEVATOR DOUBLET

SAS OFF, $V_0 = 0$

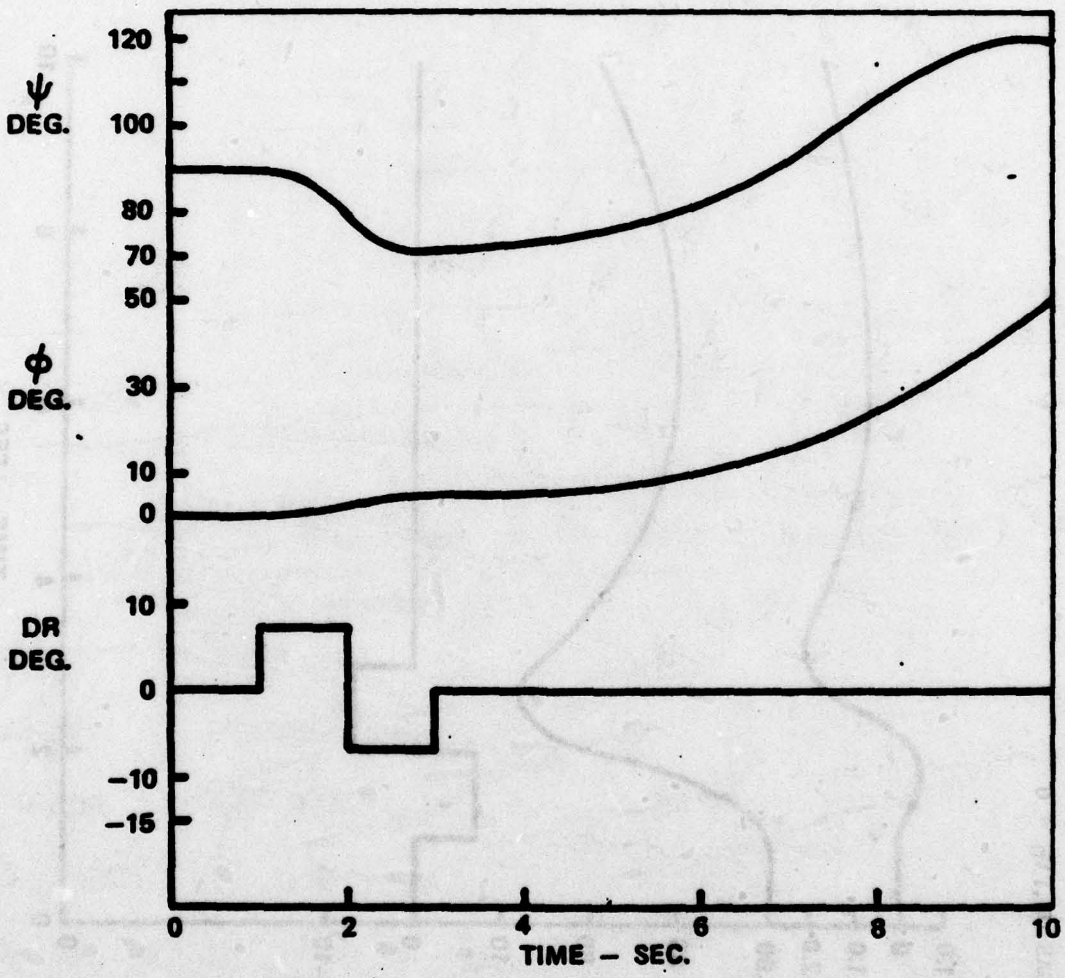


FIGURE 8. AV8A OPEN LOOP RESPONSE TO RUDDER DOUBLET

FIGURE 9. AV8A OPEN LOOP YAW RESPONSE TO RUDDER DOUBLET

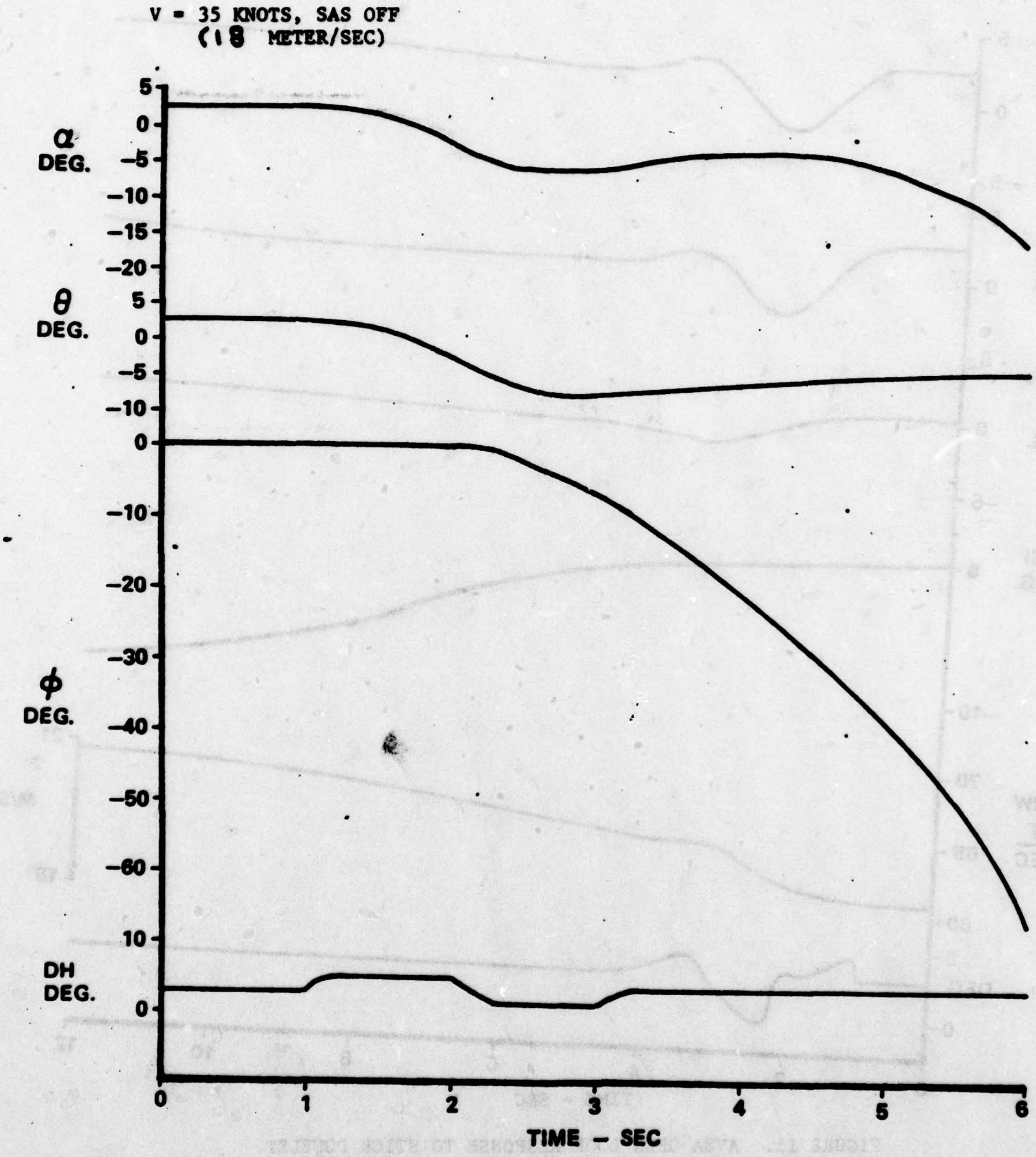


FIGURE 10. AV8A OPEN LOOP STICK DOUBLET

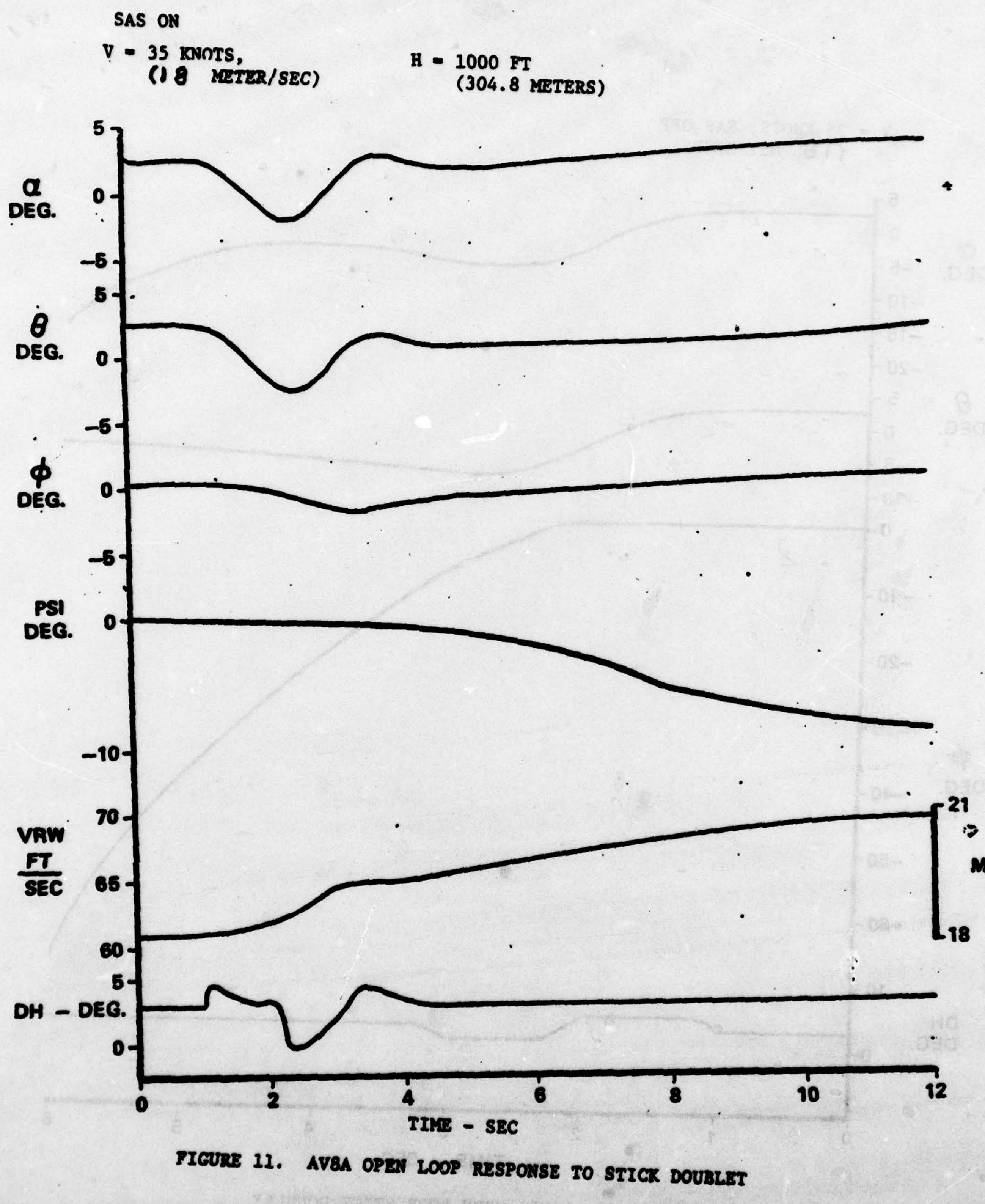


FIGURE 11. AV8A OPEN LOOP RESPONSE TO STICK DOUBLET

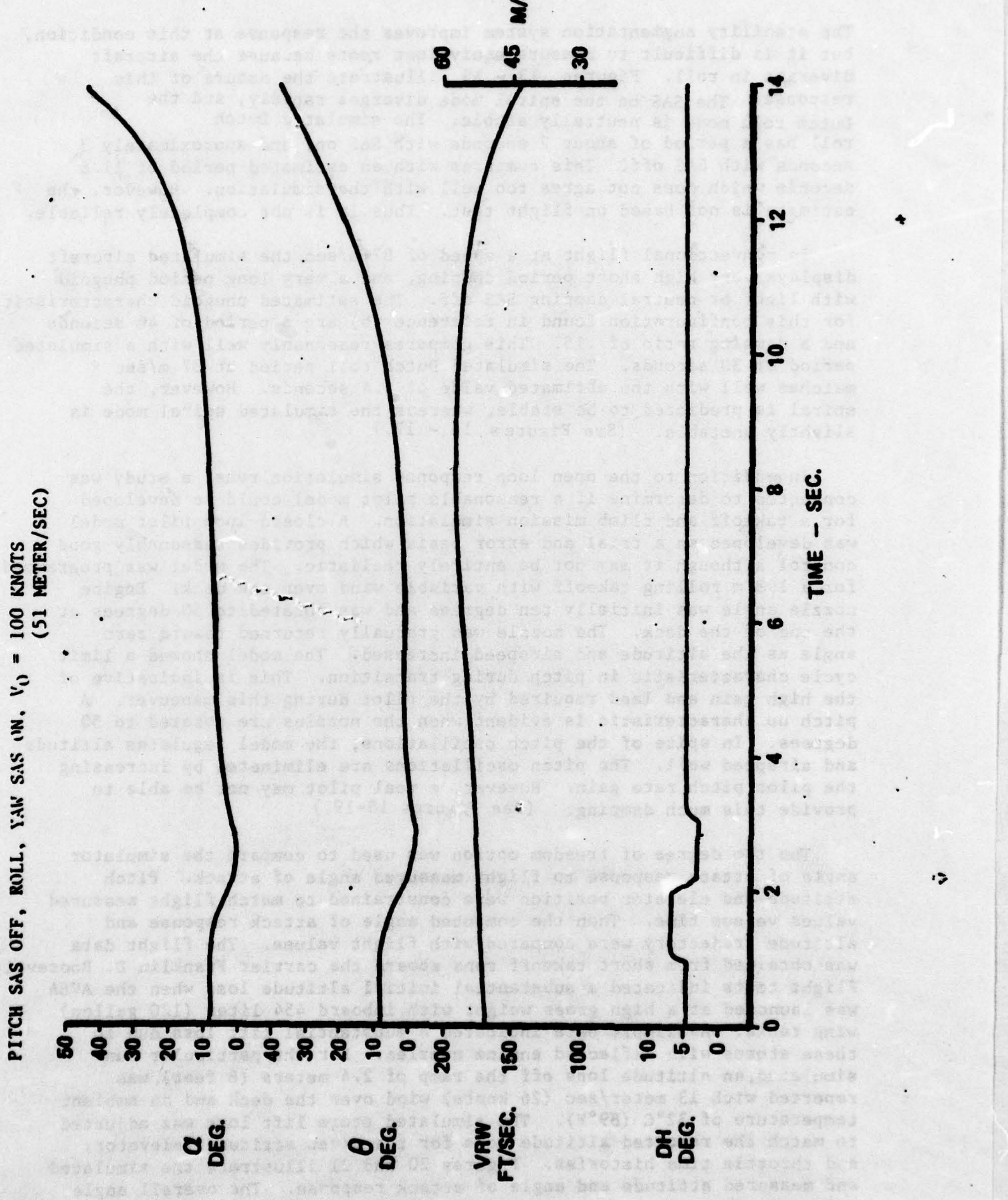


FIGURE 12. AV8A OPEN LOOP RESPONSE TO STICK DOUBLET

The stability augmentation system improves the response at this condition, but it is difficult to measure equivalent roots because the aircraft diverges in roll. Figures 13 - 15 illustrate the nature of this response. The SAS on the spiral mode diverges rapidly, and the Dutch roll mode is neutrally stable. The simulated Dutch roll has a period of about 2 seconds with SAS on, and approximately 3 seconds with SAS off. This compares with an estimated period of 11.6 seconds which does not agree too well with the simulation. However, the estimate is not based on flight test. Thus, it is not completely reliable.

In conventional flight at a speed of 87 m/sec, the simulated aircraft displays very high short period damping, and a very long period phugoid with light or neutral damping SAS off. The estimated phugoid characteristics for this configuration found in reference (b) are a period of 40 seconds and a damping ratio of .15. This compares reasonably well with a simulated period of 30 seconds. The simulated Dutch roll period at 87 m/sec matches well with the estimated value of 3.4 seconds. However, the spiral is predicted to be stable, whereas the simulated spiral mode is slightly unstable. (See Figures 16 - 17.)

In addition to the open loop response simulation runs, a study was conducted to determine if a reasonable pilot model could be developed for a takeoff and climb mission simulation. A closed loop pilot model was developed on a trial and error basis which provides reasonably good control although it may not be entirely realistic. The model was programmed for a 198 m rolling takeoff with variable wind over the deck. Engine nozzle angle was initially ten degrees and was rotated to 50 degrees at the end of the deck. The nozzle was gradually returned toward zero angle as the altitude and airspeed increased. The model showed a limit cycle characteristic in pitch during transition. This is indicative of the high gain and lead required by the pilot during this maneuver. A pitch up characteristic is evident when the nozzles are rotated to 50 degrees. In spite of the pitch oscillations, the model regulates altitude and airspeed well. The pitch oscillations are eliminated by increasing the pilot pitch rate gain. However, a real pilot may not be able to provide this much damping. (See Figures 18-19.)

The two degree of freedom option was used to compare the simulator angle of attack response to flight measured angle of attack. Pitch attitude and elevator position were constrained to match flight measured values versus time. Then the computed angle of attack response and altitude trajectory were compared with flight values. The flight data was obtained from short takeoff runs aboard the carrier Franklin D. Roosevelt. Flight tests indicated a substantial initial altitude loss when the AV8A was launched at a high gross weight with inboard 454 liter (120 gallon) wing tanks. Available data indicated a substantial lift loss due to these stores with deflected engine nozzles. For the particular run simulated, an altitude loss off the ramp of 2.4 meters (8 feet) was reported with 13 meter/sec (26 knots) wind over the deck and an ambient temperature of 32°C (89°F). The simulated store lift loss was adjusted to match the reported altitude loss for the given attitude, elevator, and throttle time histories. Figures 20 and 21 illustrate the simulated and measured attitude and angle of attack response. The overall angle

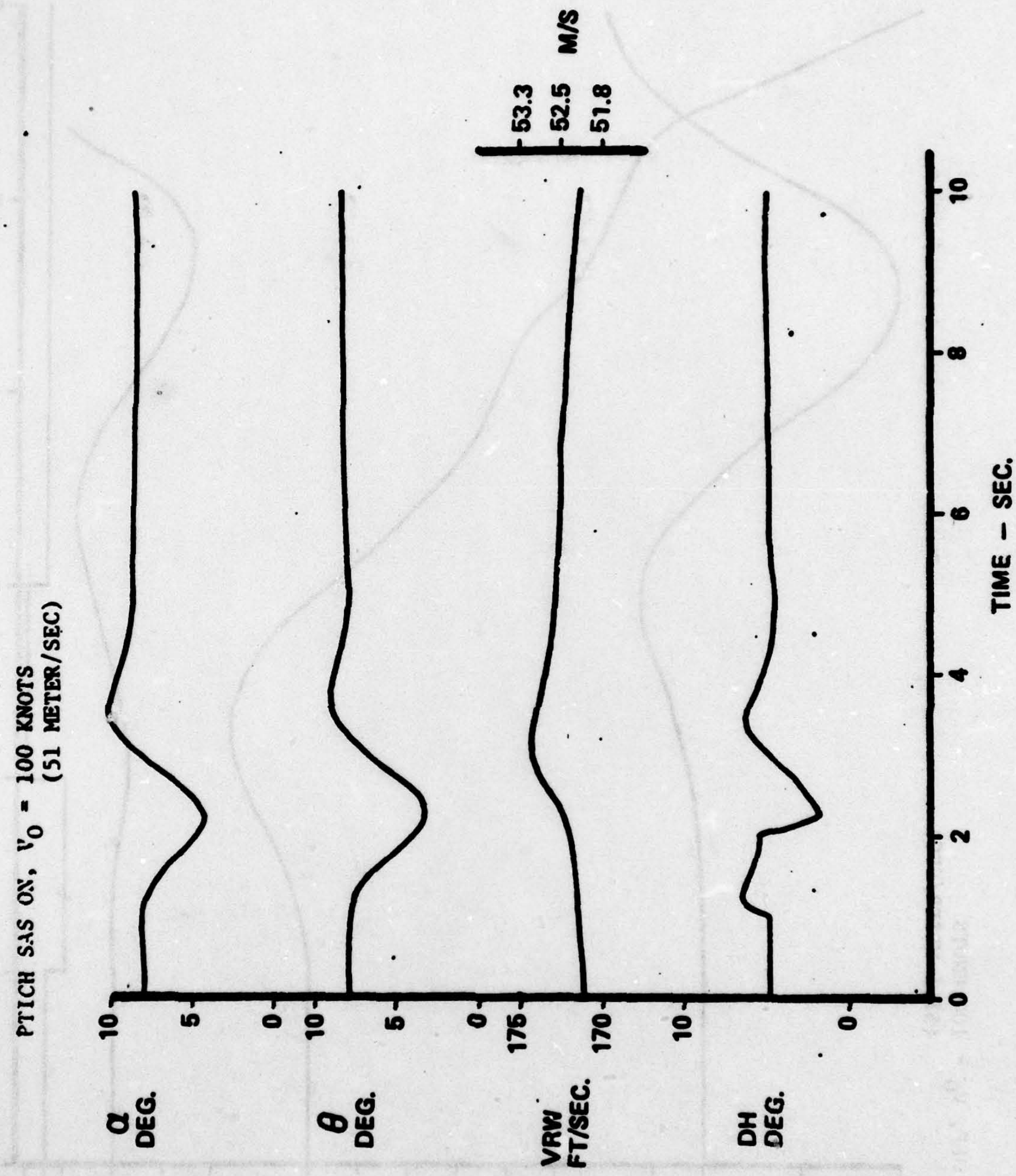


FIGURE 13. AV8A OPEN LOOP RESPONSE TO STICK DOUBLET

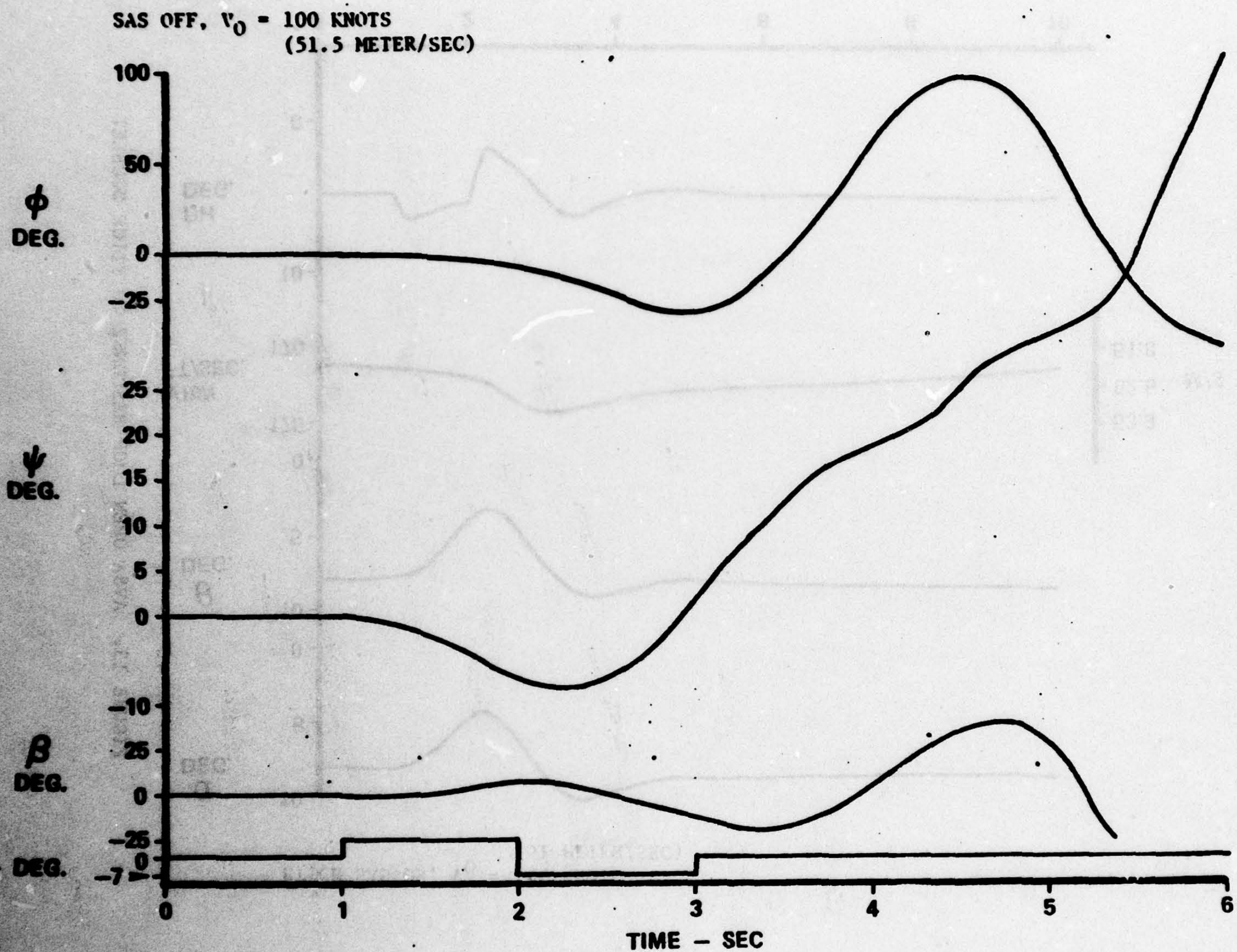


FIGURE 14. AV8A OPEN LOOP RESPONSE TO RUDDER DOUBLET

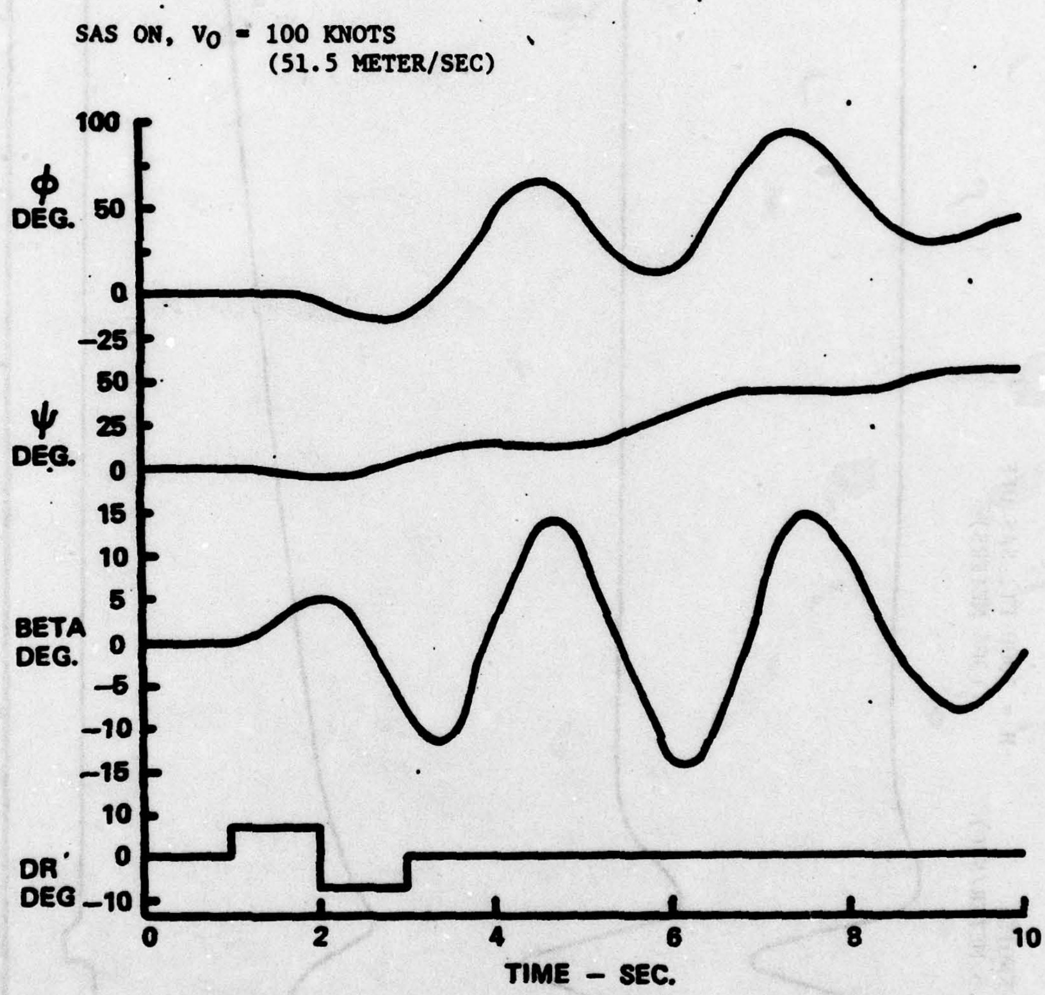


FIGURE 15. AV8A OPEN LOOP RESPONSE TO RUDDER DOUBLET

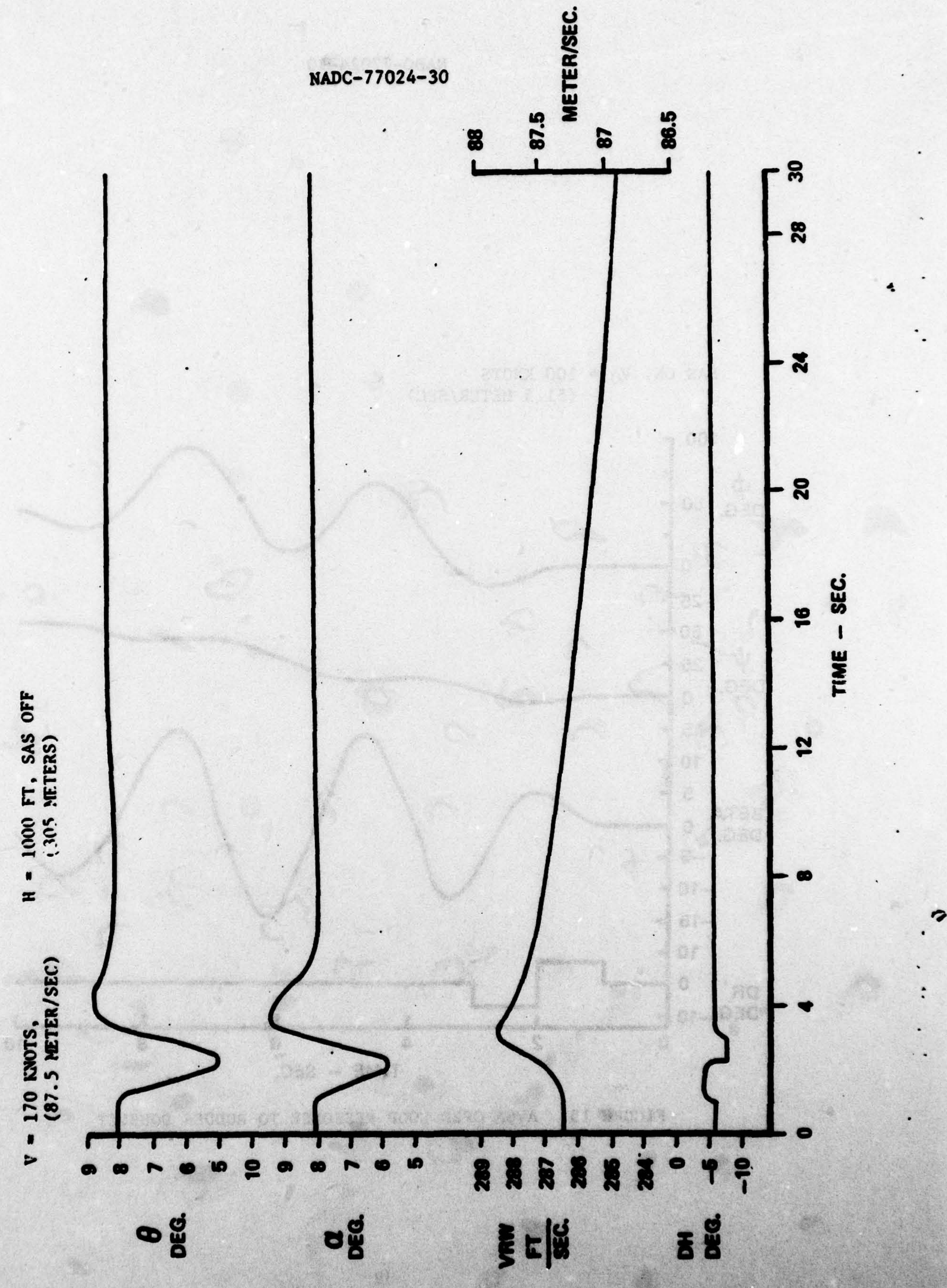


FIGURE 16. AVSA OPEN LOOP STICK DOUBLET

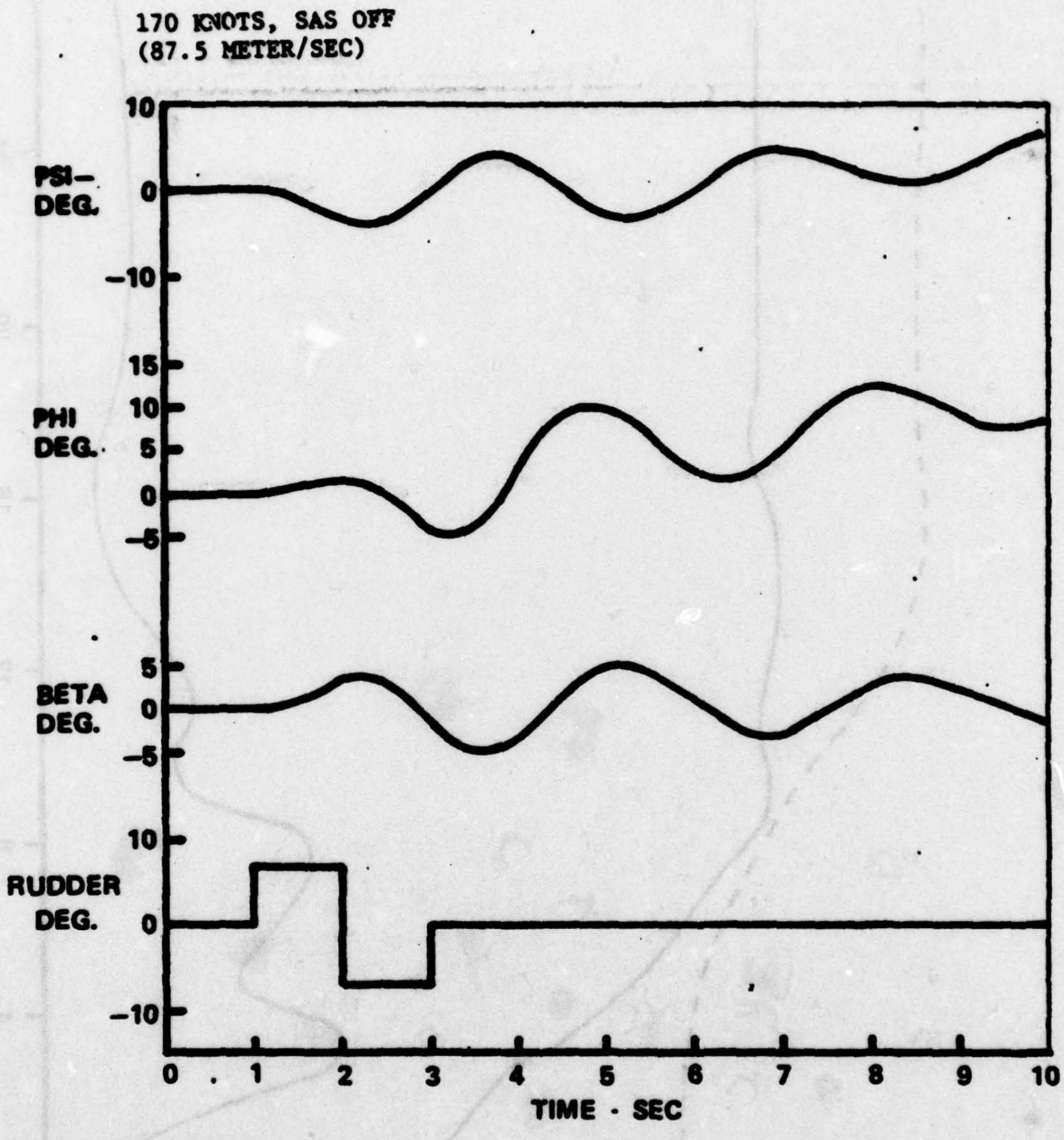


FIGURE 17. AV8A OPEN LOOP RUDDER DOUBLET

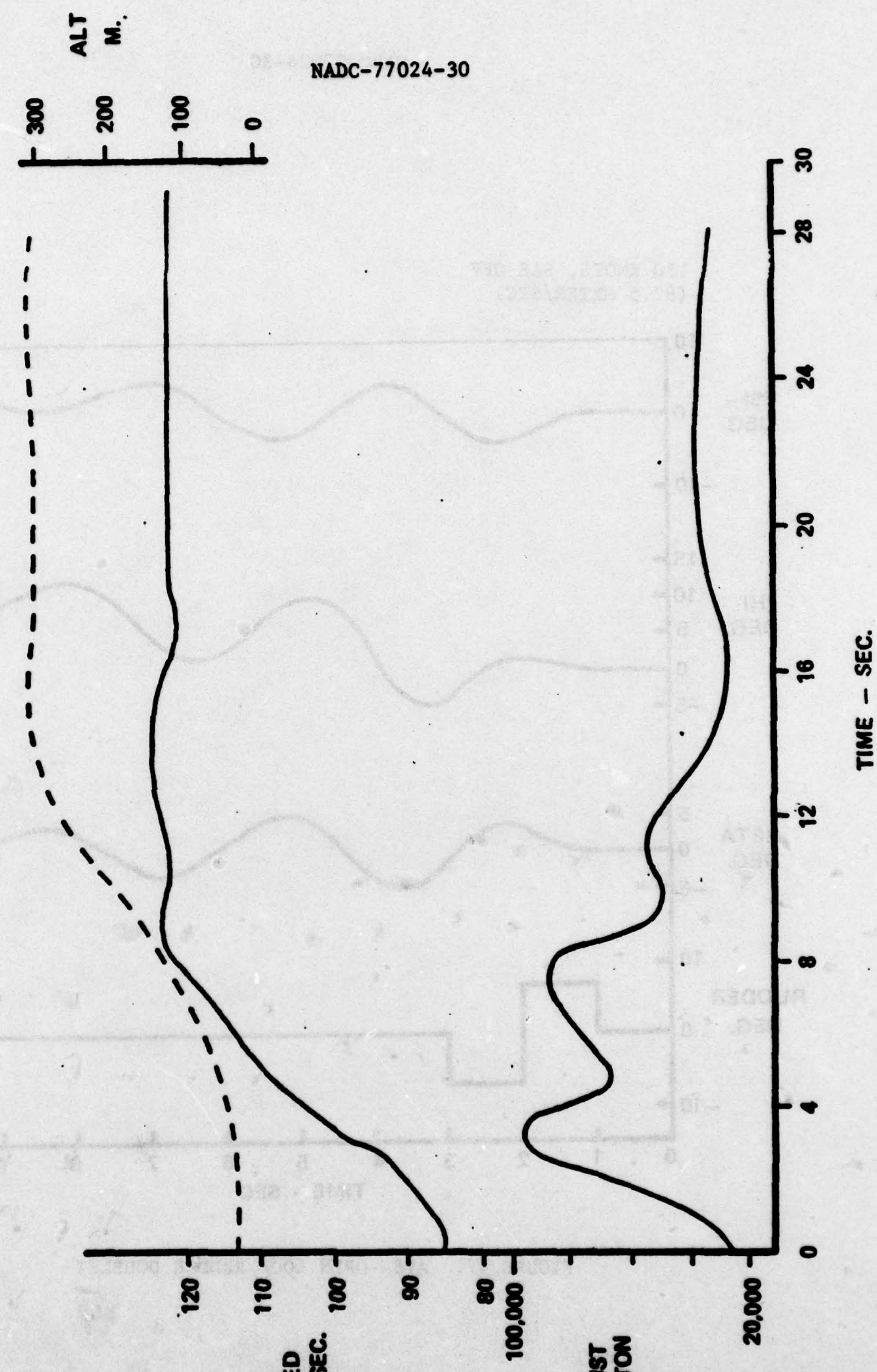


FIGURE 18. AV8A SIMULATED CLIMB RESPONSE

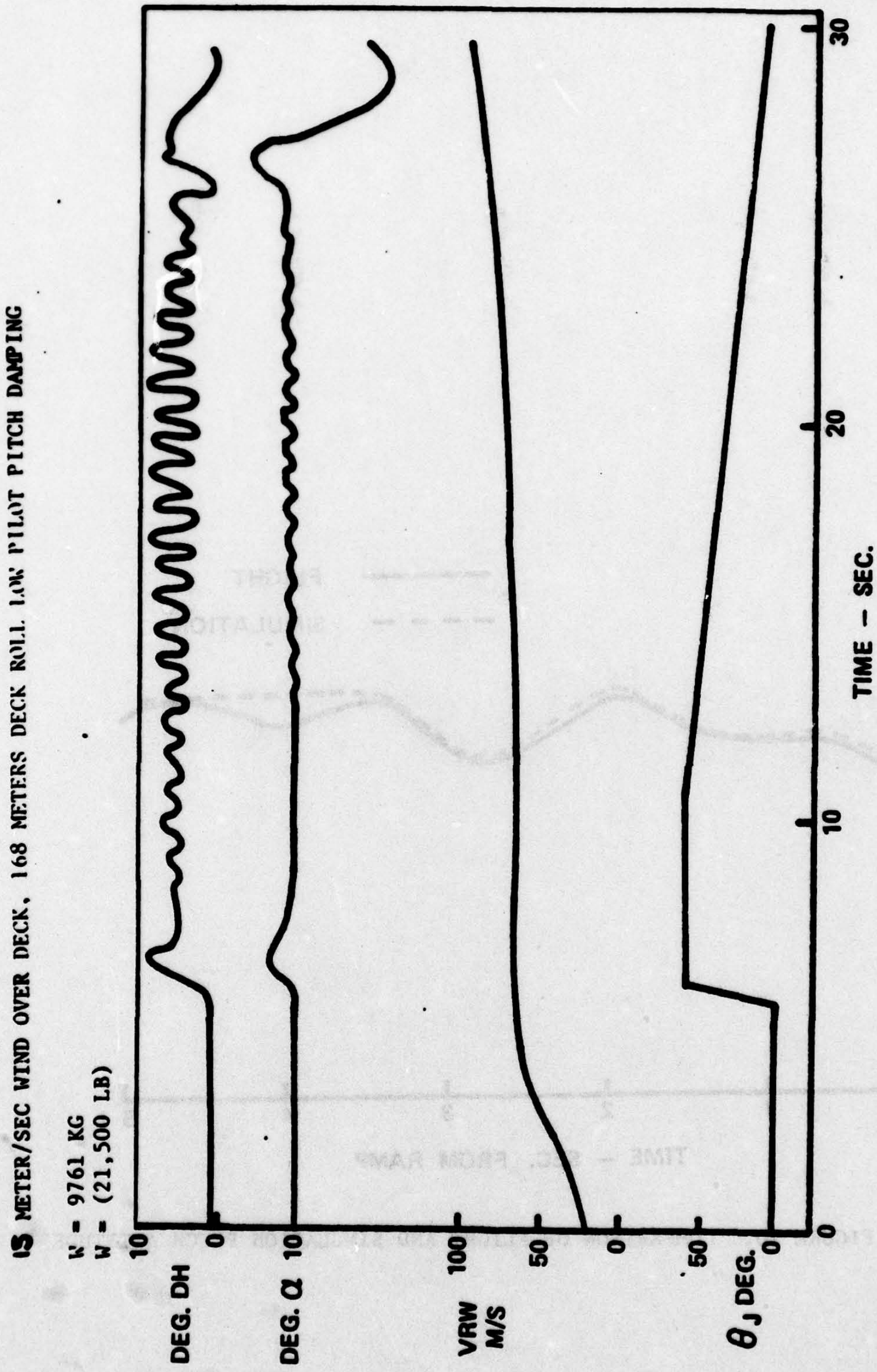


FIGURE 19. AV8A SIMULATED SHORT TAKEOFF

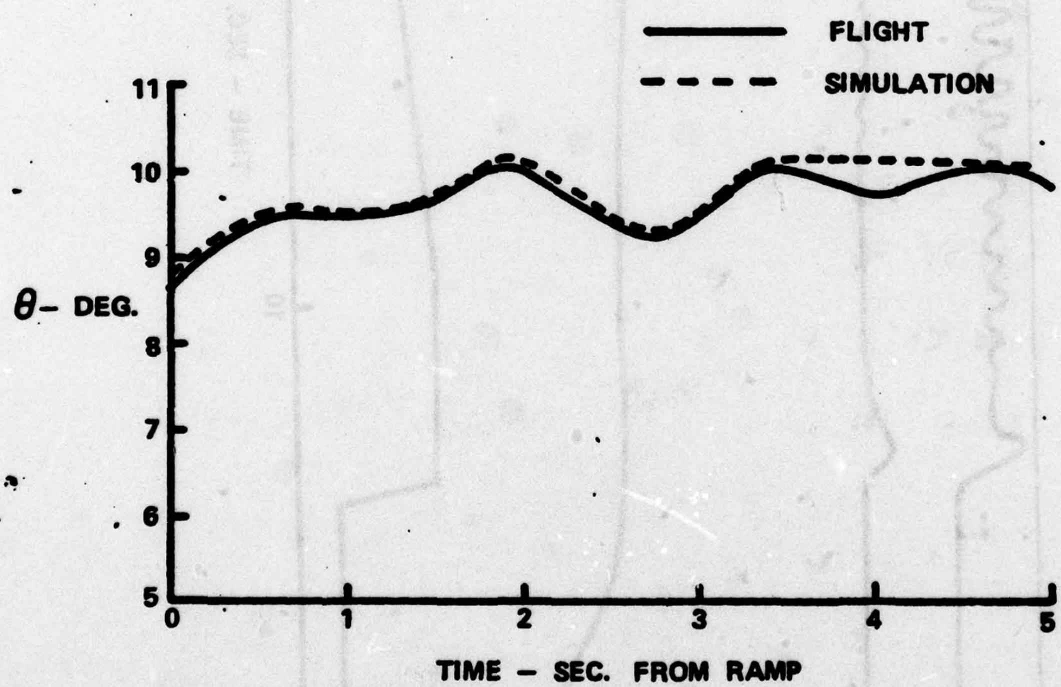


FIGURE 20. COMPARISON OF FLIGHT AND SIMULATION PITCH ATTITUDE

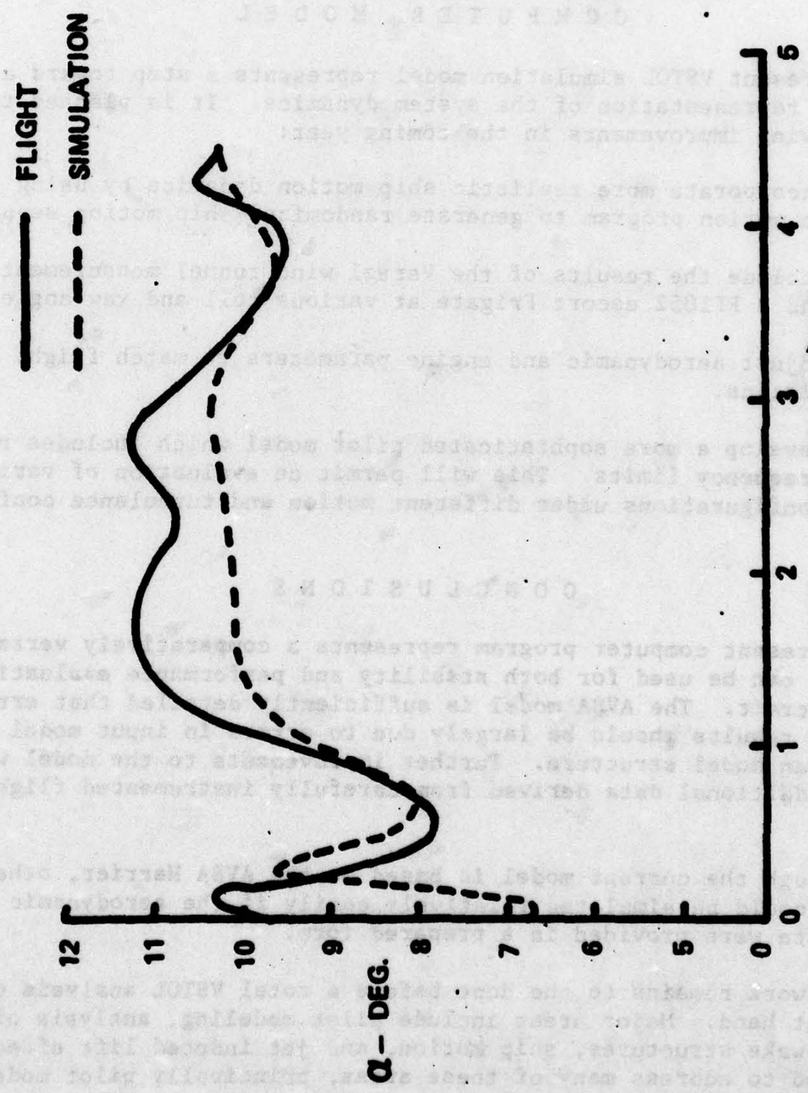


FIGURE 21. COMPARISON OF FLIGHT AND SIMULATION ANGLE OF ATTACK – DEG.

of attack match is reasonably close although not exact. There are two possible explanations to this discrepancy. First, the measured angle of attack includes distortion due to induced flow. Second, the induced lift versus airspeed may not be modeled accurately. If the measured angle of attack is assumed accurate, it indicates that the aircraft initially sinks faster than the simulated values and then arrests sink rate faster than the model. However, the actual aircraft altitude trajectory was not measured. Thus, it is not possible to determine the source of the discrepancy. Nevertheless, the simulation is capable of accurately predicting performance trends due to variation in weight, thrust, and wind over the deck.

PLANNED IMPROVEMENTS TO COMPUTER MODEL

The present VSTOL simulation model represents a step toward a truly realistic representation of the system dynamics. It is planned to make the following improvements in the coming year:

1. Incorporate more realistic ship motion dynamics by using the NSRDC ship motion program to generate randomized ship motion sequences.

2. Include the results of the Vertol wind tunnel measurement of the wake behind a FF1052 escort Frigate at various roll and yaw angles.

3. Adjust aerodynamic and engine parameters to match flight response characteristics.

4. Develop a more sophisticated pilot model which includes response lag and frequency limits. This will permit an evaluation of various control configurations under different motion and turbulance configurations.

CONCLUSIONS

The present computer program represents a comparatively versatile tool that can be used for both stability and performance evaluations of VSTOL aircraft. The AV8A model is sufficiently detailed that errors in predicted results should be largely due to errors in input model parameters rather than model structure. Further improvements to the model will require additional data derived from carefully instrumented flight tests.

Although the current model is based on the AV8A Harrier, other VSTOL aircraft could be simulated relatively easily if the aerodynamic and engine data were provided in a prepared form.

Much work remains to the done before a total VSTOL analysis capability will be at hand. Major areas include pilot modeling, analysis of turbulence and ship wake structures, ship motion, and jet induced lift effects. It is planned to address many of these areas, principally pilot modeling

and turbulence modeling in the coming fiscal year. Regular updates to the computer model will be made as additional data becomes available.

REFERENCES

(a) AV8A Aerodynamic Characteristics, Report No. MDCA1410, McDonnell Aircraft Company, 31 Jan 1972

(b) AV8A Estimated Flying Qualities, Report No. MDCA1411, McDonnell Aircraft Company, 31 Jan 1972

(c) AV8A Propulsion System, Report No. MDCA1412, McDonnell Aircraft Company, 31 Jan 1972

(d) A Study of Helicopter Landing Behavior on Small Ships, Roy M. Tuttle, Journal of the American Helicopter Society, Volume 21, Number 2, Apr 1976

(e) P-1127 (XV-6A) VSTOL Handling Qualities Evaluation, Gordon A. McKinzie, et.al., Technical Report No. 68-10, Air Force Flight Test Center, Aug 1968

(f) Introduction to the Statistical Dynamics of Automatic Control Systems - Solodovnikov, V. V., Dover Publications Inc.

(g) MIL SPEC AR-40A, Automatic Carrier Landing System Airborne Subsystem, General Requirements For

LIST OF SYMBOLS

ALPHA	angle of attack - deg
BETA	sideslip angle - deg
CD	drag coefficient
CL	lift coefficient
C_ℓ	rolling moment coefficient
$C_{\ell_{DR}}$	rolling moment coefficient per degree of rudder - 1/deg
CM	pitching moment coefficient
CN	yawing moment coefficient
CN_{DA}	yawing moment coefficient per degree of aileron - 1/deg
CN_p	yawing moment coefficient per unit roll rate - 1/rad
CY	side force coefficient
C1, C2	landing gear damping coefficient - $\frac{\text{Newton-sec}}{\text{meter}}$ $\left(\frac{\text{lb-sec}}{\text{ft}}\right)$
DH	elevator - deg
DR	rudder - deg
JPT	jet pipe temperature
K1, K2	landing gear model spring coefficients - $\frac{\text{Newton}}{\text{meter}}$ $\left(\frac{\text{lb}}{\text{ft}}\right)$
$\dot{P}$	roll acceleration - rad/sec^2
$\dot{Q}$	pitch acceleration - rad/sec^2
$\dot{R}$	yaw acceleration - rad/sec^2
RCS	reaction control system
RMS	root mean square
RPM	revolutions per minute
SAS	stability augmentation system
T/T_{max}	percent reaction control thrust

LIST OF SYMBOLS (CONT'D)

u	X body axis airspeed component - m/s (ft/sec)
$\dot{u}$	rate of change of u - m/s^2 (ft/sec^2)
UBSS	steady state horizontal burble component - m/s (ft/sec)
UB1	horizontal turbulence due to ship pitch - m/s (ft/sec)
UB2	horizontal turbulence due to random burble - m/s (ft/sec)
u/V_s	ratio of horizontal burble velocity to ship speed
V	Y body axis airspeed component - m/s (ft/sec)
$\dot{V}$	rate of change of V - m/s^2 (ft/sec^2)
VEQ	equivalent airspeed - m/s (kts)
VO	trim airspeed - m/s (ft/sec)
VRW	true airspeed - m/s (ft/sec)
VYBR	lateral burble component - m/s (ft/sec)
W	Z body axis air velocity component - m/s (ft/sec)
WBSS	steady vertical burble component - m/s (ft/sec)
WB1	vertical burble velocity due to deck pitch - m/s (ft/sec)
WB2	random component of vertical burble velocity - m/s (ft/sec)
$\dot{W}$	rate of change of aircraft Z axis velocity WRT air - m/s (ft/sec)
α	angle of attack - deg
β	angle of sideslip - deg
ΔCL	increment in lift coefficient due to stores
θ	pitch attitude - deg
μ_x	wheel braking coefficient
ϕ	bank angle - deg
ψ	yaw angle - deg

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A

COMPUTER LISTING

THIS PAGE INTENTIONALLY LEFT BLANK

```
      PROGRAM AVBA2(INPUT,OUTPUT,TAPE1,TAPE2,TAPE10,TAPE97,TAPE6=OUTPUT)
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
      COMMON/CHF/B(200)/ICHF/IB(50)
      COMMON/DAT/DATTIT(15),DATHD(40),CHTSP,SCALE(28),
     1  TMMAX,NPLOT(14),ISCALE
      COMMON/DPLOCK/IPLOT(14),IPRNT(11)
       COMMON/GAIN1/GA(100)
C
      DIMENSION NODAT(18),U(9)
        EQUIVALENCE(STAB,B(1)),(AIL,B(2)),(RUD,B(3)),(RLONSTK,B(7))
       EQUIVALENCE(RLATSTK,B(9)),(RUDPED,B(11)),(THROT,B(14)),(RN1,B(19)
     1 ),(THTN,B(17)),(TOTRM,B(20))
      EQUIVALENCE(IGRAD,IB(31))
       EQUIVALENCE(IB(22),NCHK)
       EQUIVALENCE(ISTAT,IB(42))
       EQUIVALENCE(HTIME,A(304))
       EQUIVALENCE(ITHIS, IB(40)),(IUPLOT,IB(41))
       EQUIVALENCE(A(334),DTIM),(DTIM1,A(335)),(T1,A(336))
       EQUIVALENCE(J2,IB(49)),(JLAND,IB(50))
       EQUIVALENCE(J1,IA(118))
       EQUIVALENCE(A(238),VEQIC)
     1,(IA(1),IMODE),(IA(114),IMACH),(A(241),HIC),(A(151),FG)
     2,(IA(6),IFLAT),(A(256),CG),(IA(103),ITRMP),(IA(105),NRUN)
     3,(IA(101),ICONU),(IA(23),ITPROG),(IA(61),IDT1),(IA(62),IDT2)
     4,(IA(63),IDT3),(A(303),TIME),(IA(102),ITRIM),(IA(111),ITRMCH)
      EQUIVALENCE (DT3,A(169))
      EQUIVALENCE(IA(199),N2)
       EQUIVALENCE(FREQCK,GA(18))
       EQUIVALENCE(ITD,IB(14)),(ILAND,IB(3)),(ICRASH,IB(9))
       EQUIVALENCE(IZERO,IB(4))
       EQUIVALENCE(GA(23),DTIM2)
      EQUIVALENCE(NUSED,IA(180))
C
      DIMENSION MESAGE(15)
C
       INTEGER HDATA
       DATA DTIM2/.2/
       DATA JGA/2H G/,JSCA/5HSCALE/,JNP/5HNPLOT/,JCH/5HCHTSP/,
     1  HDATA/4HHDAT/,JH/2H H/,JIB/2HIB/
      DATA JA/2H A/,JIA/2HIA/
      DATA IEND/4HEND  /
      DATA NODAT/4HSTAB,4H    ,4HAIL ,4H    ,4HDR  ,4H    ,4HLNST,4HK
     1          ,4HLTST,4HK   ,4HTHRO ,4HT   ,4HRN1 ,4H    ,4HTOTR,4HM
     2       ,4HTHTH ,4H     /
       DATA U/4HDEG.,4HDEG.,4HDEG.,4HCM. ,4HCM. ,4HDEG.,4HPCT.,4HNEWT.
     1  4HDEG./
       DATA ITIT/4HAVBA/
       DATA ITHIS/1/,IUPLOT/0/,ISTAT/0/
      DATA IUDATA/4HUDAT/
       DATA DTIM/.05/
        DATA J2/100/,JLAND/0/
       DATA DTIM1/.1/
       DATA JNUM/0/
       DATA T1/2./
      DATA IEOD/1H//
```

```
      DATA IBLANK/4H    /
      DATA IDATA/4HDATA/
      DATA IOPRN/4HOPRN/
C****************************************************************
C
C     DATA INPUT SECTION
C
00501 READ 2071,NRUN
02071  FORMAT(I2)
      NUSED=1
       IF(NRUN.EQ.0) STOP 4
00500 READ 1000,NAME,RTIME
C
C     CHECK FOR SPECIAL SYMBOLS
       N2=1
       IF(NAME.NE.IDATA) GO TO 650
C
C      PREPARE TO READ DATA
C
       PRINT 2006
C
C      READ INPUT CARDS
C
C
00540  READ 1010,IEO,NCOMMON,NCELL,VALUE,MESAGE
       IF(IEO.EQ.IEOD) GO TO 500
       GO TO (550,600,610,620,630,640,641,642) NCOMMON
C
C      FLOATING POINT COMMON
C
00550 A(NCELL) = VALUE
      PRINT 2000,JA,NCELL,VALUE,MESAGE
      GO TO 540
C
C      INTEGER COMMON
C
00600 IA(NCELL)= VALUE
      PRINT 2005,JIA,NCELL,IA(NCELL),MESAGE
      GO TO 540
00610  GA(NCELL)=VALUE
      PRINT 2000,JGA,NCELL,VALUE,MESAGE
02000  FORMAT(10X,14HDATA INPUT TO    ,A5,2H(  ,I3,2H )  ,8HCOMMON =  ,
     1 F10.4,5X,15A4)
      GO TO 540
00620  SCALE(NCELL)=VALUE
      PRINT 2000,JSCA,NCELL,VALUE,MESAGE
      GO TO 540
00630  NPLOT(NCELL)=VALUE
      PRINT 2005,JNP,NCELL,VALUE,MESAGE
02005  FORMAT(10X,14HDATA INPUT TO    ,A5,2H(  ,I3,12H )   COMMON =  ,
     1 I10,5X,15A4)
      GO TO 540
00640 CHTSM=VALUE
      PRINT 2000,JCM,NCELL,VALUE,MESAGE
      GO TO 540
```

```
00641 H(NCELL)=VALUE
      PRINT 2000,JH,NCELL,VALUE,MESAGE
      GO TO 540
00642  IH(NCELL)=VALUE
      PRINT 2005,JIH,NCELL,IH(NCELL),MESAGE
      GO TO 540
C
C*****************************************************************
C
C      CHECK IF USER IS REQUESTING INPUT
C
00650 CONTINUE
      IF(NAME.NE.MDATA) GO TO 660
00651  READ 1011,IED,NCOMMON,NCELL,VALUE,MESAGE
      IF(IED.EQ.IEOD) GO TO 500
      GO TO(652,653)  NCOMMON
00652  DATTII(NCELL)=VALUE
      PRINT 1011,IED,NCOMMON,NCELL,VALUE,MESAGE
      GO TO 651
00653  DATHD(NCELL)=VALUE
      PRINT 1011,IED,NCOMMON,NCELL,VALUE,MESAGE
01011  FORMAT(A1,I1,5X,I3,A10,15A4)
      GO TO 651
00660  CONTINUE
      IF(NAME.NE.IUDATA) GO TO 700
C
C      USER REQUESTING DATA
C
      CALL TAPRD(1,93,NG)
      GO TO 500
C
C*****************************************************************
C
C
00700 IF(NAME.NE.IOPRN) GO TO 950
      READ 704,I1READ,I2READ,ISCALE
00704  FORMAT(3(1X,I1))
      IF(I1READ.NE.0) READ 703,(IPLOT(I),I=1,14)
      IF(I2READ.NE.0) READ 703,(IPRNT(I),I=1,11)
00703 FORMAT(8I10)
00701  CONTINUE
      TIME=0.0
      IMODE=-1
      PRINT 2010
       ITPNCH=1
       ITPMP=0
       ITPROG=1
      J3=J2+10
      J1=1
07044  CONTINUE
      CALL SETUP
      IF(ITPROG.EQ.0) GO TO 800
      J1=J1+1
      IF(J1.GT.J3) GO TO 6000
      IF(J1.GT.J2.AND.IGRAD.EQ.1) GO TO 8000
```

```
      GO TO 7999
08000  CONTINUE
00800   CONTINUE
      PRINT 3000,J1
03000  FORMAT(10X,25H TRIM TERMINATED AFTER       ,I3, 11H ITERATIONS)
      N1=0
       IZERO=0
       IF(N2.EQ.1) IZERO=1
           IF(IZERO.EQ.1) CALL PRINTO
       IZERO=0
       IF(IUPLOT.EQ.1) CALL DATSAV
       RLNSTKM=2.54*RLONSTK
       RLTSTKM=2.54*RLATSTK
       THET=TOTRM*4.4416
      CALL HICHPI(ITIT,STAR,AIL,RUD,RLNSTKM,RLTSTKM,THROT,RN1,THET,
     1  THTN,NUDAT,U)
      PRINT 2007,RTIME
      RTIME1=0.
      IMODE=1
00810  CONTINUE
       IF(TIME-RTIME1) 831,832,832
00832  IF(ITHIS.EQ.1) CALL TIMEHIS(TIME)
      N1=N1+1
       DTIM=DTIM2
      IF(TIME.LT.T1.AND.NCHK.GT.0.AND.FREOCK.EQ.0) DTIM=DTIM1
       IF(NCHK.GT.0.AND.FREOCK.GT.0.)  DTIM=.25/FREOCK
      RTIME1=DTIM*N1
      IF(TIME.GE.T1.AND.NCHK.GT.0.AND.FREOCK.EQ.0) RTIME1=DTIM*N1-
     1  /DTIM1*DTIM+T1
00831  CONTINUE
      CALL LOOP2
      CALL LOOP3
       IF(IUPLOT.EQ.1) CALL DATSAV
       CALL LOOP2
      CALL LOOP3
       IF(IUPLOT.EQ.1) CALL DATSAV
      IF(TIME.GE.RTIME) GO TO 850
        IF(ITD.EQ.1.AND.JLAND.EQ.1) GO TO 850
       IF(ICRASH.EQ.1) GO TO 850
      GO TO 810
C     CHECK FOR PRINT REQUEST
00850  IF(ITHIS.EQ.1) CALL TIMEHIS(TIME)
       PRINT 851,JLAND,ITD,ICRASH
00851  FORMAT(10X,8HJLAND=    ,I2,5HITD=    ,I2,8HICRASH=    ,I2)
C
C     PRINT REQUESTED
       IF(NCHK.GT.0.OR.ISTAT.EQ.0) GO TO 702
      CALL PRINTO
00702  CONTINUE
       IF(IUPLOT.EQ.1) CALL DPLOT
      N2=N2+1
      IF(N2-NRUN) 701,701,501
C     DEAD END......COMMAND NOT RECOGNIZED
00950 PRINT 2050,NAME
      STOP 0
```

```
C     PRINT FORMAT STATEMENTS
C
C
01000 FORMAT(1X,A4,5X,G9.0)
C
01010 FORMAT(A1,I1,5X,I3,G10.0,15A4)
01020 FORMAT(A1,A3,19A4)
C
C
C
02006 FORMAT(1H1,///,20X,30(1H*),/,20X,1H*,28X,1H*,/,20X,1H*,10X
     1 , 11HINPUT DATA  ,8X,1H*,/,20X,1H*,28X,1H*,/,20X,30(1H*))
C
02007 FORMAT(1H1,///,20X,30(1H*),/,20X,1H*,28X,1H*,/,20X,1H*,10X,
     1  8HOPERATE   ,11X,1H*,/,20X,1H*,7X,10HRUN TIME  ,F7.2,5X,1H*,/
     2,20X,1H*,28X,1H*,/,20X,30(1H*))
C
02010 FORMAT(1H1,///,20X,30(1H*),/,20X,1H*,28X,1H*,/,20X,1H*,10X
     1 ,10HTRIM MODE  ,9X,1H*,/,20X,1H*,28X,1H*,/,20X,30(1H*))
C
02050  FORMAT(1H,//,10X,35HDRIVER CONTROL CARD ERROR .....      ,A4)
02060 FORMAT(1X,A1,A3,19A4)
C
C
      END
```

****

```
      SUBROUTINE SETUP
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
      COMMON/CHF/B(200)/ICHF/IB(50)
      EQUIVALENCE(A(25),ALFA),(VOIC,A(229)),(PHIIC,A(230)),(THETIC,
     1 A(231)),(PSIIC,A(232)),(HIC,A(241))
      EQUIVALENCE(IA(65),ID),(IA(111),KTRIM),(IA(102),ITRIM),
     1(ITEMP,IA(103)),(ICPRNT,IA(124)),(ITRMCM,IA(111)),(ITPROG,IA(23))
      EQUIVALENCE(INDEXT,IA(110))
      EQUIVALENCE(A(110),COURSE)
      EQUIVALENCE(STHETR,A(114)),(CTHETR,A(115))
      EQUIVALENCE(VNW,A(76)),(VEW,A(77)),(VDW,A(78))
      EQUIVALENCE(P20,A(354)),(HETAIC,B(37))
      EQUIVALENCE(STAR,B(1)),(THMSTAR,B(6)),(RLNSTKO,B(8)),
     1 (RLTSTKO,B(10)),(RPEDO,B(21)),(THROTIC,B(15)),(RLONSTK,B(7))
      EQUIVALENCE(RLATSTK,B(9)),(RUDPED,B(11)),(THROT,B(14)),
     1 (THTNIC,B(18)),(THTNC,B(16)),(RN1,B(19)),(       THTN,B(17))
      EQUIVALENCE(THRTMN,B(22)),(THRTMX,B(23)),(THTMN,B(24)),
     1(THTMX,B(25)),(PHIMN,B(26)),(PHIMX,B(27)),(PEDMN,B(28)),
     2(PEDMX,B(29)),(THTNMN,B(33)),(THTNMX,B(34))
      EQUIVALENCE(VS,B(30)),(VNWIC,B(35)),(VEWIC,B(36))
      EQUIVALENCE(HETAIC,B(37)),(STKMIN,B(38)),(STKMAX,B(39)),
     1(RLSMIN,B(40)),(PLSMAX,B(41))
      EQUIVALENCE(ITPROG,IA(23)),(TTM,A(165)),(FN,A(148)),(FE,A(149))
      EQUIVALENCE(FD,A(150)),(WAIT,A(177)),(CL,A(131)),(SPSI,A(14))
      EQUIVALENCE(CPSI,A(15)),(RHO,A(183)),(CHORD,A(182)),(AREA,A(180))
```

```
      EQUIVALENCE(OBAR,A(178)),(VHW,A(70))
      EQUIVALENCE(VN,A(64)),(VE,A(65))
      EQUIVALENCE(VEWIC,A(234))
      EQUIVALENCE(J1,IA(118))
      EQUIVALENCE(IH(19),IHIT)
      EQUIVALENCE(H(80),RLAM1),(RLAM2,H(81))
      DATA THRTMN/0./,THRTMX/102./,THTMN/-20./,THTMX/30./
      DATA PHIMN/-5./,PHIMX/5./,PEDMN/-2./,PEDMX/2./
      DATA THTNMN/0./,THTNMX/98./,STKMIN/-7.5/,STKMAX/4./,RLSMIN/-4./
      DATA RLSMAX/4./
      IHIT=0
      CTHETR    =COS(COURSE)
      STHETR=SIN(COURSE)
      VNW=-VS*CTHETR+VNWHIC
      VEW=-VS*STHETR+VEWHIC
      VNT=-VNW
      VET=-VEW
      IF(VNT)1011,1012,1011
01011 CONTINUE
      PSIWND=ATAN2(VET,VNT)*R2D
      GO TO 1013
01012 PSIWND=90.*SIGN(1.,VET)
      IF(VET.EQ.0.) PSIWND=0.
01013 CONTINUE
      PSTIC=PSIWND-BETAIC
      CALL HSETUP
      CALL LOOP2
      CALL LOOP3
      CALL LOOP2
      IF(J1.GT.2) GO TO 1014
      CALL HSETUP
      CALL LOOP2
      CALL LOOP3
      CALL LOOP2
 1014 CONTINUE
      GO TO (1001,1010,1020,1030) ,INDEXT
01001 CONTINUE
      IHIT=1
      CALL HUDIFT(2,THETIC,THTMN,THTMX,HIC,0.,100.,
     10.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.)
      RLTSTKO=0.
      RPEDO=0.
      THROT=THROTIC
      PHIIC=RLAM2*57.3
      PRINT 200,THETIC,PHIIC,HIC
 200  FORMAT(5X,8HTHETIC= ,F8.2,8HPHIIC=  ,F8.2,8HHIC=    ,F8.2  )
      GO TO 500
01010 CONTINUE
      CALL HUDIFT(4,RLNSTKO,STKMIN,STKMAX,THROTIC,THRTMN,THRTMX,
     1 RLTSTKO,RLSMIN,RLSMAX,THETIC,THTMN,THTMX,RPEDO,PEDMN,PEDMX,
     2PHIIC,PHIMN,PHIMX)
      PRINT 201,THROTIC,RLNSTKO,RLTSTKO,PHIIC,RPEDO,THETIC,PSIIC
00201 FORMAT(2X,6HTHROT= ,F8.3,6HRLN= ,F8.3,6HRLAT= ,F8.3,6HPHI=
     1 ,F8.2,6HPED= ,F8.2,6HTHET= ,F8.2,6HPSI= ,F8.2)
      GO TO 500
```

```
01020 CALL BQUIET(6,RLNSTKO,STKMIN,STKMAX,THROTIC,THRTMN,THRTMX,
     1RLTSTKO,RLSMIN,RLSMAX,THTNIC,THTNMN,THTNMX,RPEDO,PEDMN,PEDMX,
     2PHIIC,PHIMN,PHIMX)
      PHI=THROTIC
      THTN=THTNIC
      PRINT 202,RLNSTKO,THROTIC,RLTSTKO,THTNIC,RPEDO,PHIIC
00202 FORMAT(10X,   8HRLNSTKO   ,F10.5,8HTHROTIC    ,F10.5,8HRLTSTKO
     1,F10.5,  8HTHTNIC  ,F10.5, 8HRPEDO    ,F10.5,8HPHIIC    ,F10.5)
      GO TO 500
01030 VSQ=VEQIC**2
      CALL BQUIET(6,RLNSTKO,STKMIN,STKMAX,VSQ,2500.,160000.,
     1RLTSTKO,RLSMIN,RLSMAX,THROTIC,THRTMN,THRTMX,RPEDO,PEDMN,PEDMX,
     2BETAIC,-4.,4.)
      VEQIC=SQRT(VSQ)
      THROT=THROTIC
      PHI=THROT
      PRINT 203,THROTIC,RLNSTKO,VEQIC,RLTSTKO,RPEDO,BETAIC
00203 FORMAT(5X,8HTHROTIC=  ,F8.2,8HRLNSTKO=  ,F8.2,8HVEQIC=   ,F8.2,
     18HRLTSTKO=   ,F8.2,8HRPEDO=   ,F8.2,8HBETAIC=   ,F8.2)
00500 CONTINUE
      RETURN
      END
```

****

```
      SUBROUTINE BSETUP
C
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
       COMMON/CHF/B(200)/ICHF/IB(50)
C
      DIMENSION XMC(10)
       DIMENSION DXPA(15),DYPA(15),DZPA(15)
C
      EQUIVALENCE(ALFA,A(25)),(BETA,A(26)),(ALFAR,A(27)),(BETAR,A(28))
      EQUIVALENCE(ALFD,A(29)),(BETD,A(30)),(SALPH,A(31)),(CALPH,A(32))
      EQUIVALENCE(SBETA,A(33)),(CBETA,A(34))
       EQUIVALENCE(ALFAIC,B(42))
      EQUIVALENCE (PHI    ,A(1  )),(THET   ,A(2  )),(PSI    ,A(3  )),
     1            (PHIR   ,A(4  )),(THETR  ,A(5  )),(PSIR   ,A(6  )),
     2            (T11    ,A(16 )),(T21    ,A(17 )),(T31    ,A(18 )),
     3            (T12    ,A(19 )),(T22    ,A(20 )),(T32    ,A(21 )),
     4            (T13    ,A(22 )),(T23    ,A(23 )),(T33    ,A(24 )),
     5            (PB     ,A(37 )),(QB     ,A(38 )),(RB     ,A(39 )),
     6            (GAMV   ,A(35 )),(GAMH   ,A(36 ))
C
      EQUIVALENCE (VN     ,A(64 )),(VE     ,A(65 )),(VD     ,A(66 )),
     1            (VFE    ,A(67 )),(VT     ,A(68 )),(VQW    ,A(70 )),
     2            (VNB    ,A(72 )),(VEB    ,A(73 )),(VDB    ,A(74 )),
     3            (VFD    ,A(75 )),(VNW    ,A(76 )),(VEW    ,A(77 )),
     4            (VDW    ,A(78 )),(ALT    ,A(83 )),(XLON   ,A(84 )),
     5            (XLAT   ,A(85 )),(CLAT   ,A(87 )),(G      ,A(97 ))
C
      EQUIVALENCE(HPB,A(105)),(HB,A(108)),(HTV,A(109)),
```

```
1 (COURSE,A(110)),(XLATR,A(111)),(XLONR,A(112)),
2         (CLATR ,A(113)),(STHETH,A(114)),(CTHETR,A(115)),
3         (XIXX  ,A(116)),(XIYY  ,A(117)),(XIZZ  ,A(118)),
4         (XIXZ  ,A(119)),(XMC(1),A(120)),(XMASS ,A(130)),
5         (DT1   ,A(167)),(DT2   ,A(168)),(DT3   ,A(169)),
6         (WP    ,A(170)),(A(364),RHOZ)
C
  EQUIVALENCE
1         (HCG   ,A(176)),(WAIT  ,A(177)),(RHO   ,A(183)),
2         (SOUND ,A(209)),(PHIIC ,A(230)),(THETIC,A(231)),
3         (PSIIC ,A(232)),(GAMVIC,A(233)),(GAMHIC,A(234)),
4         (PBIC  ,A(235)),(QBIC  ,A(236)),(RBIC  ,A(237)),
5   .     (VEQIC ,A(238)),(XIC   ,A(239)),(YIC   ,A(240))
  EQUIVALENCE(XP,A(225)),(YP,A(226)),(ZP,A(227))
C
  EQUIVALENCE (HIC    ,A(241)),(WAITIC,A(242)),(XIXXIC,A(243)),
1             (XIYYIC,A(244)),(XIZZIC,A(245)),(XIXZIC,A(246))
  EQUIVALENCE (A(358),D2R),(A(359),R2D)
  EQUIVALENCE (A(0375),XMCC1
  EQUIVALENCE (A(0376),XMCC2 )
  EQUIVALENCE (A(0377),XMCC3 )
  EQUIVALENCE (A(0378),XMCC4 )
  EQUIVALENCE (A(0379),XMCC5 )
  EQUIVALENCE (A(0380),XMCC6 )
  EQUIVALENCE (A(0381),XMCC7 )
  EQUIVALENCE (A(0382),EXMX  )
  EQUIVALENCE (A(0383),EXMY  )
  EQUIVALENCE (A(0384),EXMZ  )
  EQUIVALENCE (UBIC,A(406)),(VBIC,A(407)),(WBIC,A(408))
C
  EQUIVALENCE (IFLAT,IA(6  )),(IDT1 ,IA(61 )),(IDT2 ,IA(62 )),
1             (IDT3 ,IA(63 )),(ICG  ,IA(64 ))
  EQUIVALENCE (ID,IA(65))
  EQUIVALENCE (IMACH,IA(114)),(XMACH,A(71))
C
  EQUIVALENCE (A(10),SPHI),(A(11),CPHI)
1,(A(14),SPSI),(A(15),CPSI)
2 ,(A(12),STHT),(A(13),CTHT)
  EQUIVALENCE (A(7),PHID),(A(8),THED),(A(9),PSID)
  EQUIVALENCE (A(43),PLH),(A(44),ULH),(A(45),RLH)
  EQUIVALENCE (IRULR,IA(104))
  EQUIVALENCE(INDEXT,IA(110))
  EQUIVALENCE(DXPA(1),A(184)),(DYPA(1),A(210)),(DZPA(1),A(257))
  EQUIVALENCE(WAITO,A(251)),(WWATIC,A(248)),(WFUELIC,A(247))
  EQUIVALENCE(WSTORE,A(252))
  DATA               OMEG/.72685E-4/,RE/2.0899E7/
C
C
C     RSETUP INITIALIZES PILOT POSITION OR AIRCRAFT C.G. POSITION AS
C
C     DESIRED. INITIALIZES LONGITUDE,LATITUDE,VN,VE,VD,VEE,VNR,VER,VDR,
C
C     VR.,WEIGHT,MASS,INERTIAS,PSIR,THETR,PHIR,PSI,THET,PHI,SINES AND
C
C     COSINES OF PSIR,THETR,PHIR,T11 THROUGH T33, AND LOOP CYCLE TIMES
```

```
C
C      IN MILLISECONDS
C
C
C      INPUTS  PSIIC,THETIC,PHIIC,GAMVIC,GAMHIC,XIC,YIC,HIC,PBIC,QBIC,
C
C              RHIC,VEQIC,WAITIC,XIXXIC,XIYYIC,XIZZIC,XIXZIC,THETRR,
C
C              XLATR,XLONR,HR,T11 THROUGH T33,XP,YP,ZP,VNW,VEW,VDW,IDT1,
C
C              IDT2,IDT3,ICG,IFLAT
C
C
C      OUTPUTS  PSI,THET,PHI,PSIR,THETR,PHIR,CLATR,STHETR,CTHETR,RH,HPR,
C
C              HCG,ALT,HTV,XLAT,XLON,CLAT,PB,QH,RB,VEQ,VT,SOUND,RHO,VN,
C
C              VE,VD,VEE,VNR,VER,VDR,VHW,WAIT,XMASS,XIXX,XIYY,XIXZ,
C
C              XMC(1) THROUGH XMC(10),DT1,DT2,DT3
C
      ID=1
C
C*****INITIALIZE PSI,THET,PHI
C
      PSI=PSIIC
      THET=THETIC
      PHI=PHIIC
      PSIR=PSI*D2R
      THETR=THET*D2R
      PHIR=PHI*D2R
C
C*****INITIALIZE SINES AND COSINES OF PSIR,THETR,PHIR + TRANSFORMATIONS
C
      CALL BTRANS
C
C***** INITIALIZE SINES AND COSINES OF THETRR
C
       STHETR=SIN(COURSE)
       CTHETR=COS(COURSE)
      CLATR=COS(XLATR)
C
C*****INITIALIZE ALT,HPR,XLAT,XLON (ICG=1 MEANS INPUT POS. I.C.S AT C.G)
C
      RH=HF+HR
      IF(ICG) 10,10,20
00010 HPR=HIC
      ALT=HPR+HP+T13*XP+T23*YP+T33*ZP
      HCG=ALT-HR
       SUMA=T11*XP+T21*YP+T31*ZP
       SUMB=T12*XP+YP+T22+ZP*T32
      XDUM=XIC-CTHETR*SUMA-STHETR*SUMB
      YDUM=YIC-CTHETR*SUMB+STHETR*SUMA
      GO TO 30
00020 HCG=HIC
```

```
      ALT=HCG+HR
       HPR=HCG-T13*XP-T23*YP-T33*ZP
      XDUM=XIC
      YDUM=YIC
00030 XLAT=(XDUM*CTHETR-YDUM*STHETR)/RR+XLATR
      XLON=(XDUM*STHETR+YDUM*CTHETR)/(CLATR*RR)+XLONR
      CLAT=COS(XLAT)
      HTV=HE+ALT
C*****INITIALIZE PB,QB,RB
C
C IF IEULER=1, THE INITIAL RATES ARE INPUT AS EULER RATES.
C PHID REPLACED BY PHIC, THED REPLACED BY QBIC, PSID REPLACED BY RBIC
C
      IF (IEULR .EQ. 0) GO TO 797
C
C ASSUMED EULER RATE SETUP
      PHID=PBIC*D2R
      THED=QBIC*D2R
      PSID=RBIC*D2R
      PB=PHID-PSID*STHT+PLB
      QB=THED*CPHI+PSID*SPHI*CTHT+QLB
      RB=PSID*CPHI*CTHT-THED*SPHI+RLB
      GO TO 798
C
00797 CONTINUE
C REGULAR INPUT
      PB=PBIC*D2R
      QB=QBIC*D2R
      RB=RBIC*D2R
 798  CONTINUE
C
C       INITIALIZE RHOZ
C
      CALL BATMOS
C
C******INITIALIZE VELOCITIES FROM EQUIVALENT AIRSPEED IN KNOTS
C*******VELOCITY INITIALIZATION*****
C
C IF  IMACH=1, THE INPUT VELOCITY VEQIC IS IN MACH NUMBERS
C
C  IF IMACH=-1 , THE VELOCITY IS INITIALIZED IN BODY AXES
C
C IF  THE AIRCRAFT IS ON THE GROUND, VEQIC IS THE GROUND (TAXI) VELOCITY
C
C IF  THE AIRCRAFT IS AIRBORNE, VEQIC IS THE EQUIVALENT VELOCITY (WITH
C   RESPECT TO THE AIR MASS).
C
C GAMVIC AND GAMHIC ARE THE INERTIAL VELOCITY VECTOR DIRECTIONAL ANGLES
C   AND, AS SUCH, IGNORE VARIATIONS IN THE WIND MAGNITUDE AND DIRECTION.
C
      STORE=0.59249*SQRT(RHO/RHOZ)
      IF (IMACH) 51,53,53
53    DUM=VEQIC*(SOUND*IMACH+(1-IMACH)/STORE)
C
C*****INITIALIZE SINES AND COSINES OF GAMV AND GAMH
```

```
C
      GAMV=GAMVIC*D2R
      SGV=SIN(GAMV)
      CGV=COS(GAMV)
      GAMH=GAMHIC*D2R
      SGH=SIN(GAMH)
      CGH=COS(GAMH)
      CB=2.*(VDW*SGV-VEW*CGV*SGH-VNW*CGV*CGH)
      CC=VNW**2+VEW**2+VDW**2-DUM**2
      AA=CB**2-4.*CC
       IF(AA)42,44,44
42    VT=-.5*CB
       GO TO 50
44     VT=.5*(-CB+SQRT(AA))
50      VN=VT*CGV*CGH
      VEE=VT*CGV*SGH
      VD=-VT*SGV
      VNR=VN-VNW
      VER=VEE-VEW
      VDR=VD-VDW
      GO TO 52
00051 VNR=T11*UBIC+T21*VBIC+T31*WBIC
      VER=T12*UBIC+T22*VBIC+T32*WBIC
      VDR=T13*UBIC+T23*VBIC+T33*WBIC
C
      VN=VNR+VNW
      VEE=VER+VEW
      VD=VDR+VDW
C
      VEQIC=STORE*SQRT(VNR**2+VER**2+VDR**2)
C
      PAD=VN**2+VEE**2
      VT=SQRT(PAD+VD**2)
      VG=SQRT(PAD)
C
      GAMVIC=0.0
      GAMHIC=0.0
      IF(VT.EQ.0.) GO TO 52
      GAMVIC=ATAN2(-VD,VG)*R2D
      IF(PAD.EQ.0.) GO TO 52
      GAMHIC=ATAN2(VEE,VN)*R2D
      SGV=SIN(GAMVIC*D2R)
      CGV=COS(GAMVIC*D2R)
      SGH=SIN(GAMHIC*D2R)
      CGH=COS(GAMHIC*D2R)
C
052    CONTINUE
       VI=SQRT(VER**2+VNR**2)
       IF(VI.EQ.0..OR.INDEXT.LT.3) GO TO 54
      GAMID=-ATAN2(VDR,VI)*57.3
       THETIC=ALFAIC+GAMID
054    CONTINUE
      IF(IFLAT) 60,60,55
55    VE=VEE
      GO TO 65
```

```
60     VE=VFE+OMEG*RTV*CLAT
65     CONTINUE
C
C
C*****INITIALIZE WAIT,MASS AND INERTIA COEFFICIENTS
C
        WAITIC=WAITO+WFUELIC+WWATIC+WSTORE
       WAIT=WAITIC
       XMASS=WAIT/32.2
       XIXX=XIXXIC
       XIYY=XIYYIC
       XIZZ=XIZZIC
       XIXZ=XIXZIC
       Z=XIXZ**2
       D=1./(XIXX*XIZZ-Z)
       XMC(1)=((XIYY-XIZZ)*XIZZ-Z)*D
       XMC(2)=(XIXX-XIYY+XIZZ)*XIXZ*D
       XMC(3)=XIZZ*D
       XMC(4)=XIXZ*D
       XMC(7)=1./XIYY
       XMC(5)=(XIZZ-XIXX)*XMC(7)
       XMC(6)=(XIXZ*XMC(7))
       XMC(9)=(XIYY-XIZZ-XIXX)*XIXZ*D
       XMC(8)=((XIXX-XIYY)*XIXX+Z)*D
       XMC(10)=XIXX*D
       XMCC1     =  XMC(4)*EXMX - XMC(3)*EXMZ
       XMCC2     =  XMC(7)*EXMX
       XMCC3     =  XMC(7)*EXMZ
       XMCC4     =  XMC(10)*EXMX - XMC(4)*EXMZ
       XMCC5     =  XMC(3)*EXMY
       XMCC6     =  XMC(4)*EXMY
       XMCC7     =  XMC(10)*EXMY
C
C*****INITIALIZE LOOP FRAME TIMES IN SECONDS
C
       DT1=.001*IDT1
       DT2=.001*IDT2
       DT3=.001*IDT3
C
       CALL HVELOC
        CALL HALFMET(DT2)
       CALL HATMOS
C
C
C       MAKE SURE THAT BLOCK DATA IS CALLED IN
C
C
       RETURN
       END

*****************************************************************************

C       TITLE                                               BQUIET
```

```
      SUBROUTINE BOUIET(NCONT,C1,CMIN1,CMAX1,C2,CMIN2,CMAX2
     1.                      C3,CMIN3,CMAX3,C4,CMIN4,CMAX4
     2.                      C5,CMIN5,CMAX5,C6,CMIN6,CMAX6)
C
C R. MCFARLAND -CSC- OCT. 19, 1972
C MODIFIED FEB 21. 1973 BY R.E. MCFARLAND
C TO (A) ESTABLISH ONE PERCENT OF RANGE AS THE PARTIAL EVAL. SIZE
C AND (B) TO MAKE THE LINEAR TRIM QUANTITIES EQUAL TO
C UHD, VHD, AND WHD RATHER THAN VND, VED, AND VDD WHICH DO NOT
C PERMIT TURNS.  IMPLEMENTED FOR 360 JUNE, 73 BY F. WATSON, CSC
C
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
       COMMON/CHF/H(200)/ICHF/IH(50)
C
C A BASIC ROUTINE FOR NULLING THE ANGULAR AND LINEAR ACCELERATION VECTOR
C BY MANIPULATION OF THE NCONT C"S (SIX IS MAXIMUM).
C
C ITPROG WILL AUTOMATICALLY RESET IF -
C ANG.ACCEL. LESS THAN .0005 RAD/SEC2
C LIN.ACCEL. LESS THAN .01 FT/SEC2
C
C      ITRMCM IS THE CONTROL SWITCH FOR THIS ROUTINE
C
C ON-THE-GROUND TRIM IS AUTOMATIC IF IHIT=1.  THE CALLING SEQUENCE, IN
C      THIS INSTANCE IS IGNOIRED
C
      DIMENSION STATE(6)
      DIMENSION HDECK(5)
       DIMENSION YZ(6),H(6,6),DX(6),V(6),HTH(6,6),W(6),LM(6),MM(6)
      DIMENSION DOC(6),DOCI(6),DOZ(6),QZ(6),QC(6)
      DIMENSION CK(6),CMIN(6)
      DIMENSION CONT(6)
      EQUIVALENCE (A(397),CONT(1))
      EQUIVALENCE (A(413),UHD),(A(414),VHD),(A(415),WHD)
      EQUIVALENCE (ITHAD,IA(166)),(IA(167),IM)
      EQUIVALENCE (IA(179),IREVAL)
C
C THE ACCEPTANCE CRITERIA ORDER IS -
C QHD,WHD,PHD,UHD,RHD,VHD
C AND ONLY THE FIRST TWO ARE USED FOR GROUND TRIM.
C
C
C
C
C THE ORDER FOR STATE IS -
C PHD,QHD,RHD,UHD,VHD,WHD
C
C
      EQUIVALENCE (IA(21),IEVAL)
      EQUIVALENCE (IGHAD,IH(31))
      EQUIVALENCE (ITRMCM,IA(111)),(ITRMP,IA(103)),(ITRIM,IA(102))
     1,(PHD,A(55)),(QHD,A(56)),(RHD,A(57)),(VND,A(88)),(VED,A(89))
     2,(VDD,A(90)),(ITPROG,IA(23)),(GAMVIC,A(233)),(PHIIC,A(230))
     3,(HIC,A(241)),(IHIT,IH(19)),(THETIC,A(231)),(XNG,A(184))
     4,(IMGT,IA(10)),(IRGT,IA(11))
```

```
C
      EQUIVALENCE (STATE(1),A(391))
      EQUIVALENCE(RLAM2,B(81)),(RLAM1,B(80))
      EQUIVALENCE(HDECK(1),A(272))
      EQUIVALENCE(XRG,A(185)),(ZRG,A(258)),(ICG,IA(64))
      EQUIVALENCE(ZP,A(227)),(XP,A(225)),(STHT,A(12)),(CTHT,A(13))
      EQUIVALENCE(ZNG,A(257)),(R2D,A(359)),(THETR,A(5))
      EQUIVALENCE(IA(110),INDEXT)
C
      DATA GAIN/1./,NHALF/10/
      DATA INEW/0/
      DATA WSCALE1/1./,WSCALE2/1./
      DATA DQZ/0.0005,0.01,0.0005,0.01,0.0005,0.01/
      DATA XIJK/.4/
C
      DATA DWCI/6*.01/
      D1 = UBD
      D2 = VBD
      D3 = WBD
      IF(ITWMCM) 10,10,60
00010 IF(ITMPE) 50,50,20
00020 IF(IM) 160,160,30
00030 CONTINUE
      ITRMP=0
00040 STATE(1)=PBD
      STATE(2)=QBD
      STATE(3)=RBD
      STATE(4) = D1
      STATE(5) = D2
      STATE(6) = D3
      IF(INIT) 41,41,48
00041 GO TO (47,46,45,44,43,42),N
00042 CONT(6)=C6
00043 CONT(5)=C5
00044 CONT(4)=C4
00045 CONT(3)=C3
00046 CONT(2)=C2
00047 CONT(1)=C1
00048 CONTINUE
00050 CONTINUE
      IF(IPEVAL.EQ.0) INEW=0
      RETURN
C
00060 CONTINUE
      IF(ITMMP) 70,70,63
00063 IF(IHIT-IHITP) 70,160,70
C
00070 DO 72 I=1,6
      W(I)=DQZ(1)/DQZ(I)
00072 CONTINUE
      GAIN=1.
      IGRAD=0
      ISTEP=0
C
      IF(INIT)  120,120,80
```

```
C GROUND TRIM SETUP
00080 GAMVIC=0.
       PHIIC=RLAM2*R2D
      INITP=1
      N=2
      M=2
      ZX=SQRT((ZNG-ZRG)**2+(XNG-XRG)**2)
       THETR=ATAN2((ZNG-ZRG),ZX)+RLAM1
      THETIC=R2D*THETR
      STHT=SIN(THETR)
      CTHT=COS(THETR)
      IF(ICG)  82,82,84
00082 HICMAX=(ZNG-ZP)*CTHT+(XP-XRG)*STHT+HDECK(1)+1.
      CMIN(2)=HICMAX-3.
      GO TO 86
00084  HICMAX=ZNG*CTHT-XRG*STHT+HDECK(1)+1.
      CMIN(2)=HICMAX-3.
00086 CK(2)=HICMAX-CMIN(2)
       Z1=ZNG-ZRG+1.
       THTMAX=ATAN2(Z1,SQRT(Z1**2+(XNG-XRG)**2))*R2D +RLAM1*R2D
       Z2=ZNG-ZRG-1.
       CMIN(1)=ATAN2(Z2,SQRT(Z2**2+(XNG-XRG)**2))*R2D +RLAM1*R2D
      CK(1)=THTMAX-CMIN(1)
      DOC(1)=.01
      DOC(2)=.01
      GO TO 140
C NORMAL TRIM SETUP
00120 N=NCONT
      M=6
      INITP=0
      DO 130 I=1,6
00130 DOC(I)=DOCI(I)
00132 GO TO (138,137,136,135,134,133),N
00133 CK(6)=CMAX6-CMIN6
      CMIN(6)=CMIN6
00134 CK(5)=CMAX5-CMIN5
      CMIN(5)=CMIN5
00135 CK(4)=CMAX4-CMIN4
      CMIN(4)=CMIN4
00136 CK(3)=CMAX3-CMIN3
      CMIN(3)=CMIN3
00137 CK(2)=CMAX2-CMIN2
      CMIN(2)=CMIN2
00138 CK(1)=CMAX1-CMIN1
      CMIN(1)=CMIN1
00140 ITRMP=1
      IEVAL=0
      JJ=0
C ESTABLISH STATE VECTOR
00160 GO TO (166,165,164,163,162,161),M
00161 Q7(6) = D2
00162 Q7(5)=HHD
00163 Q7(4) = D1
00164 Q7(3)=PHD
00165 Q7(2) = D3
```

```
00166 QZ(1)=QHD
      PRINT 502,(QZ(I),I=1,6)
00502  FORMAT(10X,7HQZ(I)=    ,6G13.5)
C ACCEPTANCE
C
      ITPROG=1
      DO 204 I=1,M
      IF(ABS(QZ(I))-DQZ(I)) 204,204,206
00204 CONTINUE
      ITPROG=0
00206 CONTINUE
       IF(ITPROG.EQ.0) RETURN
00180 GO TO (186,185,184,183,182,181),N
00181 QC(6)=(C6-CMIN(6))/CK(6)
00182 QC(5)=(C5-CMIN(5))/CK(5)
00183 QC(4)=(C4-CMIN(4))/CK(4)
00184 QC(3)=(C3-CMIN(3))/CK(3)
00185 QC(2)=(C2-CMIN(2))/CK(2)
00186 QC(1)=(C1-CMIN(1))/CK(1)
00190 IF(JJ) 200,200,220
00200  IF(IGRAD.EQ.1.AND.ISTEP.LE.25) GO TO 201
       DO 210 I=1,M
00210  YZ(I)=QZ(I)
       ERSQP=0.
       DO 218 I=1,M
00218 ERSQP=ERSQP+(YZ(I)*W(I))**2
        GO TO 203
00201  ERSQ=0.
       DO 202 J=1,M
00202  ERSQ=ERSQ+(QZ(J)*W(J))**2
       ERSQP1=.9*ERSQP
       IF(ERSQ.GT.ERSQP1) IGRAD=0
      IF(IGRAD.EQ.0) ISTEP=0
       PRINT 515, ERSQ,ERSQP
00515  FORMAT(10X,6HERSQ=    ,G13.5,10H ERSQP=    ,G13.5)
       IF(IGRAD.EQ.0) GO TO 200
        ERSQP=ERSQ
       GO TO 337
00203  CONTINUE
      IF(IHIT) 209,209,217
00209 GO TO (216,215,214,213,212,211),N
00211 CONT(6)=C6
00212 CONT(5)=C5
00213 CONT(4)=C4
00214 CONT(3)=C3
00215 CONT(2)=C2
00216 CONT(1)=C1
00217 CONTINUE
      STATE(1)=PHD
      STATE(2)=QHD
      STATE(3)=RHD
      STATE(4) = D1
      STATE(5) = D2
      STATE(6) = D3
C
```

```
      JJ=1
C
C
      IEVAL=1
C
      GO TO 310
00220 DO 230 I=1,M
00230 H(I,JJ)=(QZ(I)-YZ(I))/DQC(JJ)
C
      QC(JJ)=QC(JJ)-DQC(JJ)
C
      IF(INEW) 240,240,290
00240 IEVAL = 0
C
00290 CONTINUE
C
      IF(JJ-N) 300,320,320
C
00300 JJ=JJ+1
      IEVAL=0
C
00310 QC(JJ)=QC(JJ)+DQC(JJ)
      IM=0
C
      GO TO 390
C
00320 CONTINUE
      JJ=0
      INEW=1
      IEVAL=0
      IM=1
      DO 330 L=1,N
      DO 330 J=1,N
      HTH(L,J)=0.
      DO 330 I=1,M
00330 HTH(L,J)=HTH(L,J)+H(I,L)*H(I,J)*W(I)
C
       PRINT 503,((H(I,J),I=1,6),J=1,6)
00503  FORMAT(10X,8HH(I,J)=    ,/,6(6G13.5,/))
       CALL MINV(HTH,N,DET,LM,MM)
       IGRAD=1
       GAIN=1.
      IF(INDEXT.EQ.1) GAIN=.5
C
00337  CONTINUE
      IF(ISTEP.GT.0) GO TO 338
      GO TO 336
00338  DO 339 J=1,6
00339  YZ(J)=QZ(J)
00336  CONTINUE
      DO 340 I=1,N
      V(I)=0.
      DO 340 J=1,M
00340 V(I)=V(I)-H(J,I)*YZ(J)*W(J)
C
```

```
      DO 350 I=1,N
      DX(I)=0.
      DO 350 J=1,N
00350 DX(I)=DX(I)+HTH(I,J)*V(J)
       ISTEP=ISTEP+1
      IF(ISTEP.GE.25) IGRAD=0
       IF(IGRAD.EQ.0) ISTEP=0
       IF(INDEXT.EQ.1) GO TO 351
       GO TO 352
00351 CONTINUE
         DET=H(1,1)*H(2,2)-H(2,1)*H(1,2)
      DX(2)=(1.*YZ(1)*H(1,2)-YZ(2)*H(1,1))/(DET+1.)
      DX(1)=(-YZ(1)*H(2,2)+1.*YZ(2)*H(2,1))/(DET+1.)
      PRINT 353,DX(1),DX(2),YZ(1),YZ(2)
00353 FORMAT(1X,7HDX(1)=   ,F10.5,7HDX(2)=   ,F10.5,7HYZ(1)=   ,F10.5,
     1  7HYZ(2)=   ,F10.5)
00352  CONTINUE
C
C
C XIJK PERCENT MAXIMUM EXCURSION
C
      RSCALE1=1.
      RSCALE2=1.
      XIJK=.4
      IF(INDEXT.EQ.1) XIJK=.2
00360 DO 370 I=1,N
      XX=DX(I)
      IF(XX-XIJK) 363,363,362
00362  RSCALE1=XIJK/XX
       GO TO 369
00363  IF(XX+XIJK) 364,369,369
00364  RSCALE1=-XIJK/XX
00369  CONTINUE
      IF(RSCALE1.LT.RSCALE2) RSCALE2=RSCALE1
00370  CONTINUE
      IF(IGRAD.EQ.1) GAIN=GAIN*.97
      DO 371 I=1,N
00371  OC(I)=OC(I)+DX(I)*GAIN*RSCALE2
C
00390 DO 395 I=1,N
      XY=OC(I)
      IF(XY) 392,392,393
00392 OC(I)=0.05
      GO TO 395
00393 IF(XY-1.0) 395,394,394
00394 OC(I)=0.95
00395 CONTINUE
      IF(IMIT) 396,400,396
00396 HIC=OC(2)*CK(2)+CMIN(2)
      TWETIC=OC(1)*CK(1)+CMIN(1)
      PRINT 397,OC(1),OC(2)
00397  FORMAT(1X,7HOC(1)=   ,F10.5,1X,7HOC(2)=   ,F10.5)
      GO TO 407
00400 CONTINUE
      GO TO (406,405,404,403,402,401),N
```

```
00401 C6=QC(6)*CK(6)+CMIN(6)
00402 C5=QC(5)*CK(5)+CMIN(5)
00403 C4=QC(4)*CK(4)+CMIN(4)
00404 C3=QC(3)*CK(3)+CMIN(3)
00405 C2=QC(2)*CK(2)+CMIN(2)
00406 C1=QC(1)*CK(1)+CMIN(1)
00407 CONTINUE
      RETURN
      END

*****************************************************************************

      SUBROUTINE LOOP2
C
C MODIFIED BY R.E. MCFARLAND    JAN.14,1974
C    ***********************************************************
C    * LOOP2 IS A COMBINATION OF FSAA BASIC SLOOP1 AND SLOOP2  *
C    *                                                         *
C    * LOOP2 CALLS THE FOLLOWING SUBROUTINES---                *
C    * NOTE:  SUBROUTINES IN  PARENTHESES HAVE BEEN CODED IN-LINE
C    *    CONTR2                                               *
C    *    AERO2                                                *
C    *    WINDC                                                *
C    *    (RTORQUE)                                            *
C    *    RROTATE                                              *
C    *    (RFTOTAL)                                            *
C    *    RTRANSFO                                             *
C    *    (RVERTICA)                                           *
C    *    (RHORIZON)                                           *
C    *    RVELOCIT                                             *
C    *                                                         *
C    *    (RACCELERATION)                                      *
C    ***********************************************************
C    *                                                         *
C    * IF NUSED=1, THIS LOOP USES DT1 AS THE FRAME TIME        *
C    *          2                 DT2                          *
C    *                                                         *
C    ***********************************************************
C    * AUTHOR: D. JONES -CSC-        APRIL 1973                *
C    ***********************************************************
C
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
       COMMON/CHF/H(200)/ICHF/IB(50)
       DIMENSION DXPA(15),DYPA(15),DZPA(15)
       DIMENSION ITOUCH(5),VDKN(7),VDKE(7)
       EQUIVALENCE(VR,A(454))
       EQUIVALENCE(ITOUCH(1),IA(10)),(IHRAK,IB(29)),(VDKN(1),A(433))
      EQUIVALENCE(FGXM,A(455))
       EQUIVALENCE(VDKE(1),A(440))
C
C
C
      EQUIVALENCE (CPHI   ,A(011)),(STHT   ,A(012)),(CTHT   ,A(013))
```

```
C
      EQUIVALENCE (T11    ,A(16 )),
000001             (T21    ,A(17 )),(T31    ,A(18 )),(T12    ,A(19 )),
000002             (T22    ,A(20 )),(T32    ,A(21 )),(T13    ,A(22 )),
000003             (T23    ,A(23 )),(T33    ,A(24 ))
C
      EQUIVALENCE (PH    ,A(37)),
000002       (QB    ,A(38 )),(RB    ,A(39 )),(PBD    ,A(55 )),
000003       (QBD   ,A(56 )),(RBD   ,A(57 )),
000004       (VN    ,A(64 )),(VE    ,A(65 )),(VD    ,A(66 )),
000005       (VEE   ,A(67 ))
      EQUIVALENCE(DXPA(1),A(184)),(DYPA(1),A(210)),(DZPA(1),A(257))
      EQUIVALENCE(SPSI,A(14)),(CPSI,A(15))
      EQUIVALENCE(XP,A(225)),(YP,A(226)),(ZP,A(227))
      EQUIVALENCE(IGEAR,IA(104))
       EQUIVALENCE(WAIT,A(177))
      EQUIVALENCE(ISTAT,IB(42))
      EQUIVALENCE(NCHK,IB(22))
C
      EQUIVALENCE (ALTD   ,A( 80)),(ALT    ,A( 83)),(SLAT   ,A( 86)),
000002             (CLAT   ,A( 87)),(VND    ,A( 88)),
000003             (VED    ,A(89 )),(VDD    ,A(90 )),(AX     ,A(91 )),
000004             (AY     ,A(92 )),(AZ     ,A(93 )),(AXP    ,A(94 )),
000005             (AYP    ,A(95 )),(AZP    ,A(96 )),(G      ,A( 97)),
000006   (RTV,A(109)),(XMASS,A(130))
C
      EQUIVALENCE (FAX,A(136)),(FAY,A(137)),(FAZ,A(138)),
000001             (FEX,A(139)),(FEY,A(140)),(FEZ,A(141)),
000002             (FGX,A(142)),(FGY,A(143)),(FGZ,A(144)),
000003             (FTX,A(145)),(FTY,A(146)),(FTZ,A(147))
C
      EQUIVALENCE (FN    ,A(148)),(FE    ,A(149)),(FD    ,A(150)),
000001             (FG    ,A(151))
C
      EQUIVALENCE (TAL   ,A(155)),(TAM   ,A(156)),(TAN   ,A(157)),
000001             (TEL   ,A(158)),(TEM   ,A(159)),(TEN   ,A(160)),
000002             (TGL   ,A(161)),(TGM   ,A(162)),(TGN   ,A(163)),
000003             (TTL   ,A(164)),(TTM   ,A(165)),(TTN   ,A(166))
C
      EQUIVALENCE (DT1   ,A(167)),(DT2   ,A(168))
C
       EQUIVALENCE(HR,A(170))
       EQUIVALENCE(HCG,A(176)),(IMODE,IA(1)),(IFLAT,IA(6)),(IFFCI,IA(7))
       EQUIVALENCE(ITRIM,IA(102)),(NRUN,IA(105)),(NUSED,IA(180))
        EQUIVALENCE(VX,A(200)),(VY,A(201))
       EQUIVALENCE(WFUEL,A(244)),(WWAT,A(250)),(
000001  WAITO,A(251)),(WSTORE,A(252))
C
C
C
      DATA RATE/16.1/
      DATA TS/0.0/,OMEG/.72685E-4/
C
C
      DT1=DT2
```

```
      DELTH=.5*DT2
C
C
C
C***********************************************************************
C
C                 ***************************************
C                 * CALL FAST CONTROL SUBROUTINE        *
C                 ***************************************
C
C                 * CALL FAST AERO SUBROUTINE
C                 ***************************************
C
C
       CALL PILOT
      CALL CONTR2
       CALL ENGINE
      CALL AERO2
C                 ***************************************
C                 ***************************************
C
C                 * CALL TURBULENCE SUBROUTINE          *
C                 ***************************************
C
      CALL WINDC
C
C     ***********************************************************************
C     * THE FOLLOWING IS THE EQUIVALENT OF -BTORQUE- IN-LINE CODED        *
C     ***********************************************************************
C     *                                                                   *
      IF(IGEAR) 1,1,2
01    FGX=0.
      FGY=0.
      FGZ=0.
      TGL=0.
      TGM=0.
      TGN=0.
      GO TO 3
02    CALL GEARS
03    CONTINUE
C     * THIS PORTION CALCULATES TOTAL TORQUE IN BODY AXES(L,M, AND N)     *
C     *                                                                   *
C     * INPUTS- TAL,TAM,TAN,TEL,TEM,TEN,TGL,TGM,TGN                       *
C     *                                                                   *
C     * OUTPUTS-TTL,TTM,TTN                                               *
C     ***********************************************************************
C
C
      TTL=TAL+TEL+TGL
      TTM=TAM+TEM+TGM
      TTN=TAN+TEN+TGN
C     ***********************************************************************
C
C
C                 ***************************************
C                 *  CALL SUBROUTINE TO CALCULATE RATATIONAL  *
```

```
C               *  ACCELERATIONS,VELOCITIES, AND ANGLES      *
C               ********************************************
C
      CALL BROTATE(DT2)
C
C
C     **********************************************************************
C     * THE FOLLOWING IS THE EQUIVALENT OF -BFTOTAL- CODED IN-LINE         *
C     **********************************************************************
C     *                                                                    *
C     * PORTION CALCULATES TOTAL FORCES IN BODY AXES                       *
C     *                                                                    *
C     * INPUTS- FAX,FAY,FAZ,FEX,FEY,FEZ,FGX,FGY,FGZ                        *
C     *                                                                    *
C     * OUTPUTS- FTX,FTY,FTZ                                               *
C     **********************************************************************
C
      FTX=FAX+FEX+FGX
      FTY=FAY+FEY+FGY
      FTZ=FAZ+FEZ+FGZ
C     **********************************************************************
C
C               ********************************************
C               * CALL SUBROUTINE TO CALCULATE STANDARD     *
C               * TRANSFORMATIONS, AND SINES AND COSINES    *
C               * PHI, THETA, AND PSI                       *
C               ********************************************
C
      CALL BTRANS
C
C
C
C     **********************************************************************
C     * THE FOLLOWING IS THE EQUIVALENT OF -BVERTICA- IN-LINE CODED        *
C     **********************************************************************
C     *                                                                    *
C     * THIS PORTION CALCULATES FD AND VDD, INTEGRATES VDD TO GET VD,      *
C     * CALCULATES ALTD, INTEGRATES ALTD TO GET ALT, CALCULATES HCG,HWEEL*
C     *                                                                    *
C     * INPUTS- FTX,FTY,FTZ,T13,T23,T33,FG,RTV,VE,VN,XMASS,HR,IFLAT,IMODE*
C     *                                                                    *
C     * OUTPUTS- VDD,VD,ALTD,ALT,HCG,FD,HWEEL                              *
C     **********************************************************************
C     * C
C     * C
      FD=T13*FTX+T23*FTY+T33*FTZ
       WAIT=WAITO+WWAT+WFUEL+WSTORE
        XMASS=WAIT/G
      VDD=(FD+FG)/XMASS
      IF(IFLAT) 20,10,20
C
C     ROTATING EARTH
C
00010 VDD=-(VE**2+VN**2)/RTV+VDD
C
```

```
00020 IF(IMODE) 30,90,40
C
00030 VDDP=VDD
      VDP=VD
      DELTH=0.5*DTI
      GO TO 90
C
00040 IF(IFFCI) 50,50,90
0050  VD=VD + DELTH*(3.0*VDD-VDDP)
      VDDP=VDD
      ALT=ALT - DELTH*(VD+VDP)
      VDP=VD
  90   ALTD=-VD
      HCG=ALT-HR
C
C
C
C     ***************************************************************
C     * THE FOLLOWING IS EQUIVALENT TO -BHORIZON- IN-LINE CODED      *
C     ***************************************************************
C     *                                                             *
C     * THIS PORTION CALCULATES FN,FE,VND,VED,INTEGRATES TO GET VN,VE, *
C     * CALCULATES VEE                                              *
C     *                                                             *
C     *                                                             *
C     * INPUTS- T11,T21,T31,T12,T22,T32,FTX,FTY,FTZ,XMASS,RTV,SLAT,CLAT, *
C     * IMODE,IFLAT,IFFCI                                           *
C     *                                                             *
C     * OUTPUTS- FN,FE,VND,VED,VN,VE,VEE                            *
C     ***************************************************************
      FN=T11*FTX+T21*FTY+T31*FTZ
      FE=T12*FTX+T22*FTY+T32*FTZ
      TEMP=1./XMASS
      IF(IFLAT) 200,100,200
C
C*****ROTATING EARTH
C
100   TLAT=SLAT/CLAT
      TEMP1=1./RTV
      VND=FN*TEMP+(VN*VD-VE**2*TLAT)*TEMP1
      VED=FE*TEMP+(VE*VD+VN*VE*TLAT)*TEMP1
      GO TO 300
C
C*****FLAT EARTH
C
200   VND=FN*TEMP
      VED=FE*TEMP
300   IF(IMODE) 400,700,500
00400 VNDP=VND
      VEDP=VED
      GO TO 700
500   IF(IFFCI) 600,600,700
00600 VN=VN + DELTH*(3.0*VND-VNDP)
      VNDP=VND
      VE=VE + DELTH*(3.0*VED-VEDP)
```

```
      VEDP=VED
700   VEF=VE
      FTX1=FEX+FAX-WAIT*STHT
      IF(ABS(FTX1).GT.ABS(FGXM)) GO TO 702
       IF(IBRAK.EQ.1.AND.ITOUCH(2).EQ.1.AND.ABS(VR).LT.2.) GO TO 701
      GO TO 702
0701   VN=VDKN(1)
       VE=VDKE(1)
       VEE=VE
0702   CONTINUE
      IF(IFLAT) 900,800,900
C
C*****ROTATING EARTH
C
800   VEE=VE-OMEG*RTV*CLAT
900   CONTINUE
       VX=VN*CPSI+VE*SPSI
       VY=VE*CPSI-VN*SPSI
       VDR=VDKN(1)*CPSI+VDKE(1)*SPSI
      VR=VX-VDR
C
C             **************************************
C             * CALL SUBROUTINE TO CALCULATE INERTIAL VEL.*
C             * W.R.T. INERTIAL WIND COMPONENTS, AND       *
C             * BODY AXIS VELOCITEIS U,V, AND W            *
C             **************************************
C
      CALL BVELOC
C
C             **************************************
C             * CALL SUBROUTINE TO CALCULATE QUANTITIES    *
C             *   RELATIVE TO ALPHA AND BETA               *
C             **************************************
C
      CALL BALFBET(DT2)
C
C
C
C
C
C     ***************************************************************
C     * THE FOLLOWING IS EQUIVALENT TO -BACCELERATION- IN-LINE CODED *
C     ***************************************************************
C     *                                                              *
C     * THIS PORTION  CALCULATES BODY AXIS CG, AND PILOT ACCELERATIONS *
C     *                                                              *
C     * INPUTS- VDD,XMASS,FG,T11,T21,T31,T12,T22,T32,T13,T23,T33,VND,VED,*
C     *                                                              *
C     *         PH,QH,RH,PBD,QBD,RBD,XP,YP,ZP                        *
C     *                                                              *
C     * OUTPUTS- AX,AY,AZ,AXP,AYP,AZP                                *
C     ***************************************************************
C
C
      AX=FTX/XMASS
```

```
      AY=FTY/XMASS
      AZ=FTZ/XMASS
C
C*****PILOT ACCELERATION
C
       XP=DXPA(15)-DXPA(5)
       YP=DYPA(15)-DYPA(5)
       ZP=DZPA(15)-DZPA(5)
      AXP=AX-(RB**2+QB**2)*XP+(PB*QB-RBD)*YP+(PB*RB+QBD)*ZP
      AYP=AY+(PB*QB+RBD)*XP-(RB**2+PB**2)*YP+(QB*RB-PBD)*ZP
      AZP=AZ+(PB*RB-QBD)*XP+(QB*RB+PBD)*YP-(QB**2+PB**2)*ZP
C
C
C              ****************************************
C              * CALL FAST USER UTILITY SUBROUTINE    *
C              ****************************************
C
      IF(ISTAT.EQ.0) GO TO 3000
      CALL SHOW
03000  CONTINUE
C
      RETURN
C
      END

*****************************************************************************
      .

      SUBROUTINE LOOP3
C
C     D. JONES - COMPUTER SCIENCES CORPORATION
C
C
C
C MODIFIED BY R.E. MCFARLAND    JAN.14,1974
C
C
C
C
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
C
      EQUIVALENCE (A(81),XLOND),(A(82),XLATD),(A(84),XLON),
     2            (XLAT  ,A(85 )),(SLAT  ,A(86 )),(CLAT  ,A(87 )),
     3            (HTV   ,A(109)),(FG    ,A(151)),(WAIT  ,A(177))
C
      EQUIVALENCE (T11   ,A(16 )),(T21   ,A(17 )),(T31   ,A(18 )),
     1            (T12   ,A(19 )),(T22   ,A(20 )),(T32   ,A(21 )),
     2            (T13   ,A(22 )),(T23   ,A(23 )),(T33   ,A(24 )),
     3            (XLON  ,A(84 )),(XLAT  ,A(85 )),(HWEEL ,A(102)),
     4            (XPR   ,A(103)),(YPR   ,A(104)),(HPR   ,A(105)),
     5            (DNR   ,A(106)),(DER   ,A(107)),(RR   ,A(108))
C
      EQUIVALENCE (XLATR ,A(111)),(XLONR ,A(112)),(CLATR ,A(113)),
     1   (STHETR,A(114)),(CTHETR,A(115)),(XCG,A(174)),
```

```
3               (YCG    ,A(175)),(HCG    ,A(176)),
3               (GAMV   ,A(35 )),(GAMH   ,A(36 )),(PL     ,A(40 )),
4               (QL     ,A(41 )),(RL     ,A(42 )),(PLB    ,A(43 )),
5               (QLB    ,A(44 )),(RLB    ,A(45 )),(VN     ,A(64 ))
C
      EQUIVALENCE (VE     ,A(65 )),(VD     ,A(66 )),(VEE    ,A(67 )),
1               (VT     ,A(68 )),(VG     ,A(69 )),(VNR    ,A(72 )),
2               (VER    ,A(73 )),(VDR    ,A(74 )),(SLAT   ,A(86 )),
3               (CLAT,A(87)),(RTV,A(109)),(DT3,A(169))
C
      EQUIVALENCE (IFLAT,IA(6))
      EQUIVALENCE (IMODE,IA(1)),(IFFCI,IA(7))
      EQUIVALENCE (INALG3,IA(89))
     1,( A(83) , ALT)
      EQUIVALENCE (A(130),XMASS)
       EQUIVALENCE(XP,A(225)),(YP,A(226)),(ZP,A(227))
      EQUIVALENCE (NUSED,IA(180))
      EQUIVALENCE (DT2,A(168))
C
C
      DATA TS/0./,RE/20898908./
C
C
C
C
C
C
C
      NUSED=1
      DT1=DT2*NUSED
       IFFCI=0
C
C
      CALL BATMOS
C
C*****THIS PORTION OF THE PROGRAM CALCULATES RADIUS TO CG,
C
C*****GRAVITY FORCE,LATITUDE AND LONG-
C
C*****ITUDE DOT, INTEGRATES TO GET LATITUDE AND LONGITUDE, AND CALCULATE
C
C*****SINE AND COSINE OF LATITUDE
C
C
C*****INPUTS- WAIT,VN,VEE,ALT,IMODE
C
C
C*****OUTPUTS-RTV,FG,XLATD,XLOND,XLAT,XLON,SLAT,CLAT
C
C
C*****RADIUS AND GRAVITY FORCE
C
      RTV=ALT+RE
      FG=WAIT*(RE/RTV)**2
      G=FG/XMASS
```

```
C
C*****LATITUDE AND LONGITUDE
C
      XLATD=VN/RTV
      XLOND=VEE/(CLAT*RTV)
      IF(IMODE) 10,30,20
00010 XLONDP=XLOND
      XLATDP=XLATD
      DELTH=0.5*DTI
      GO TO 30
00020 IF(IFFCI.EQ.1) GO TO 30
      XLON=XLON + DELTH*(XLOND+XLONDP)
      XLONDP=XLOND
      XLAT=XLAT + DELTH*(XLATD+XLATDP)
      XLATDP=XLATD
00030 SLAT=SIN(XLAT)
      CLAT=COS(XLAT)
C
C
C*******************************************************************
C
C*****THIS PORTION OF THE PROGRAM CALCULATES NORTH AND
C
C             YP
C*****EAST POSITION OF CG. X AND Y POSITION
C
C
C*****INPUTS-RP,XLAT,XLATR,XLONR,T11,T12,T13,T21,T22,T23,T31,T32,T33,XP,
C
C*****ZP,CLATR,STHETR,CTHETR,HPR
C
C
C*****OUTPUTS-DNR,DER,XCG,YCG,XPR,YPR,HPR
C
C
C*****NORTH AND EAST POSITION OF C.G. FROM RUNWAY (0,0) COORDINATES
C
      DNR=RR*(XLAT-XLATR)
      DER=RR*CLATR*(XLON-XLONR)
C
C      TITLE    JAN 6, 1973                              LOOP3
C*****X AND Y POSITION OF C.G. W.R.T. RUNWAY
C
      XCG=DNR*CTHETR+DER*STHETR
      YCG=-DNR*STHETR+DER*CTHETR
C
C*****X,Y, AND H POSITION OF PILOT W.R.T. RUNWAY
C
      DNPR=DNR+T11*XP+T21*YP+T31*ZP
      DEPR=DER+T12*XP+T22*YP+T32*ZP
      XPR=DNPR*CTHETR+DEPR*STHETR
      YPR=-DNPR*STHETR+DEPR*CTHETR
      HPR=HCG-T13*XP-T23*YP-T33*ZP
C
C*******************************************************************
```

```
C
C*****THIS PORTION OF THE PROGRAM CALCULATES INERTIAL P.Q. AND R
C
C*****DUE TO NON-FLAT EARTH, TRUE AIR
C
C***SPEED,GROUND VELOCITY, FLIGHT PATH ANGLES IN VERTICAL AND HORIZONTAL
C
C*****INERTIAL PLANE
C
C
C*INPUTS- VN,VE,VD,VEE,RTV,SLAT,CLAT,T11,T12,T13,T21,T22,T23,T31,T32,T33
C
C*****          VNR,VER,VDR
C
C
C*****OUTPUTS- PL,QL,RL,PLB,QLB,RLB,VT,VG,GAMV,GAMH
C
C
C*****INERTIAL P,Q,R  (IFLAT=1 MEANS FLAT EARTH-OTHERWISE ROTATING)
C
      IF(IFLAT) 100,100,200
00100 PL=VE/RTV
      QL=-VN/RTV
      RL=-VE*SLAT/(RTV*CLAT)
      PLB=T11*PL+T12*QL+T13*RL
      QLB=T21*PL+T22*QL+T23*RL
      RLB=T31*PL+T32*QL+T33*RL
      GO TO 300
00200 PL=0.0
      QL=0.
      RL=0.
      PLB=0.
      QLB=0.
      RLB=0.
C
C*****FLIGHT PATH ANGLES, GROUND VELOCITY AND TRUE AIRSPEED
C
00300 PAD=VN**2 +VEE**2
      VT=SQRT(PAD+VD**2)
      VG=SQRT(PAD)
      IF(VT.EQ.0.0) GO TO 350
      GAMV=ATAN2(-VD,VG)
      IF(PAD.EQ.0.0) GO TO 400
      GAMH=ATAN2(VEE,VN)
00350 CONTINUE
00400 CONTINUE
C
C*********************************************************************
C
C*****THIS PORTION OF THE PROGRAM CALLS THE USER ROUTINES FOR
C
C*****THE THIRD LOOP ENGINE,CONTROL SYSTEM,ENGINE AND SPECIAL
C
C*****UTILITY FUNCTIONS
C
```

```
C
C
C
C
C
C
C
C
C
C
      CALL DECK
      RETURN
      END

**************************************************************************


      SUBROUTINE BATMOS
C
C     ****************************************************************
C     * BATMOS CALCULATES TOTAL VELOCITY RELATIVE TO THE WIND,       *
C     *                                                              *
C     *                                                              *
C     * AIR DENSITY, MACH NO., DYNAMIC AND IMPACT PRESSURES,         *
C     *                                                              *
C     * CALIBRATED AND EQUIVALENT AIRSPEEDS, SPEED OF SOUND          *
C     *                                                              *
C     *                                                              *
C     * INPUTS - ALT,UB,VB,WB                                        *
C     *                                                              *
C     * OUTPUTS - VRW,RHO,XMACH,QBAR,QBARC,VCAL,SOUND                *
C AS OF APRIL, 1974 NO LONGER OUTPUTS VRW.  SEE BALFBET
C     *                                                              *
C     ****************************************************************
C     * UPDATED 3 DEC 73 TO REFLECT CHANGES MADE TO CURRENT FSAA VERSION *
C     ****************************************************************
C
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
C
      EQUIVALENCE (A(058),UB     ),(A(059),VB     ),(A(060),WB     ),
     2      (A(070),VRW    ),(A(071),XMACH ),(A(075),VEQ    ),
     3      (A(083),ALT    ),(A(101),VCAL   ),(A(178),QBAR   ),
     4      (A(179),QBARC ),(A(183),RHO    ),(A(209),SOUND  ),
     5      (A(332),TR     ),(A(333),PR     ),(A(364),RHOZ   ),
     6      (A(365),RRHOZ ),(A(366),TAMB   ),(A(367),PAMB   ),
     7      (A(368),TTOT  ),(A(369),PTOT   ),(A(370),DELAT  )
      EQUIVALENCE(A(371),SQTEMPR)
      EQUIVALENCE (VNR,A(72)),(VER,A(73)),(VDR,A(74))
C
      EQUIVALENCE (IA(001),IMODE ),(IA(141),ICUND )
C
      DATA HLEV/36089./,XM1/1./,PPESZ/2116.2/,TEMPZ/518.69/
C     ****************************************************************
C     * "VRW" IS TOT. AIRCRAFT VELOCITY W.R.T. WIND                  *
```

```
C     ****************************************************************
C     ****************************************************************
C     * MODE CONTROL...(-1) IS "I.C.", (0) IS "HOLD", (1) IS "OPERATE"  *
C     ****************************************************************
C
      IF(IMODE) 2,3,3
C
    2 CALL ARDC62(0.0,SOUNDZ,RHOZ)
C
C     ****************************************************************
C     * "ICOND" IS CONSTANT DENSITY SELECTION SWITCH                  *
C     *                                                               *
C     *  IF "ICOND" SET TO (1), ALL CALCULATIONS ARE REFERENCED TO    *
C     * THE DENSITY AT THE ALTITUDE SPECIFIED BY "HRHOZ"              *
C     ****************************************************************
C
    3 IF(ICOND)4,4,6
C
    4 XX=ALT
      GO TO 7
C
    6 XX=HRHOZ
C
    7 CONTINUE
      CALL ARDC62(XX,SOUND,RHO)
C
C
C     ****************************************************************
C     * "TR" IS THE TOT. TEMPERATURE RATIO                            *
C     * "PR" IS THE TOT. PRESSURE RATIO                               *
C     ****************************************************************
C
      TR=1.+.2*XMACH**2
C
      IF(XMACH-XM1) 11,11,12
C
00011 PR=TR*TR*TR*SQRT(TR)
      GO TO 13
C
00012 IF(XMACH) 121,121,122
C
00121 PR=1.0
      GO TO 13
C
00122 CONTINUE
      TEMP1=XMACH**2
      TEMP2=7.-1./TEMP1
      PR=166.9*TEMP1/(TEMP2*TEMP2*SQRT(TEMP2))
C
00013 IF(ALT-HLEV) 40,40,50
C
C     ****************************************************************
C     * "TAMB" IS AMBIENT TEMPERATURE IN DEGREES KELVIN               *
C     * "PAMB" IS AMBIENT PRESSURE IN LB/FT2                          *
C     * "TAMBR" IS RATIO OF "TAMB" AT ALTITUDE TO SAME AT SEA LEVEL   *
```

```
C    * "PAMBR" IS RATIO OF "PAMB" AT ALTITUDE TO SAME AT SEA LEVEL    *
C    * "PTOT" IS TOTAL PRESSURE                                       *
C    * "TTOT" IS TOTAL TEMPERATURE                                    *
C    ******************************************************************
C
00040 TAMBR=1.-6.875E-6*ALT
      PAMBR=TAMBR**5.256
      GO TO 60
C
00050 TAMBR=.751895
      PAMBR=.2234*EXP(-4.806E-5*(ALT-HLEV))
C
00060 CONTINUE
      TAMB=DELAT+TAMBR*TEMPZ*.5555555
      PAMB=PAMBR*PRESZ
C
      PTOT=PR*PAMB
      TTOT=TR*TAMP
C
C    ******************************************************************
C    * "RHO" IS DENSITY OF AIR AT ALTITUDE                            *
C    * "SOUND" IS SPEED OF SOUND AT ALTITUDE                          *
C    *                                                                *
C    * "RHO" AND "SOUND" ARE MODIFIED BY DELTA TEMPERATURE EFFECT     *
C    *     "DELAT" IS THE DELTA TEMPERATURE DIFFERENCE OF DESIRED     *
C    *     AMBIENT FROM STANDARD TEMPERAURES IN DEGREES CENTIGRADE    *
C    ******************************************************************
C
       YCON=TAMB/(TAMB-DELAT)
      SQTEMPR=SQRT(YCON)
C
      RHO=RHO/YCON
      SOUND=SOUND*SQRT(YCON)
C
C    ******************************************************************
C    *  "XMACH" IS MACH NO.                                           *
C    *  "QBAR" IS DYNAMIC PRESSURE                                    *
C    *  "QBARC" IS IMPACT PRESSURE                                    *
C    *  "VEQ" IS EQUIVALENT AIRSPEED IN KNOTS                         *
C    *  "VCAL" IS CALIBRATED AIRSPEED IN KNOTS                        *
C    ******************************************************************
C
      XMACH=VRW/SOUND
C
      QBAR=.5*RHO*VRW**2
C
      QBARC=PTOT-PAMB
C
      VEQ=.59249*VRW*SQRT(RHO/RHOZ)
C
      IF (VEQ .LT. 10.)  GO TO 70
      VCAL=.59249*SOUNDZ*SQRT(5.0*((QBARC/PRESZ+1.0)**.2857-1.0))
      GO TO 71
C
00070 CONTINUE
```

```
      VCAL  = VEO
C
00071 RETURN
      END

*********************************************************************************

      SUBROUTINE BROTATE(DELT)
C
C MODIFIED BY R.E. MCFARLAND    JAN.14,1974
C
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
       COMMON/CHF/B(200)/ICBF/IB(50)
C
      DIMENSION XMC(10)
       DIMENSION ITOUCH(5)
C
       EQUIVALENCE(ITOUCH(1),IA(10))
      EQUIVALENCE (PHI   ,A(1  )),(THET  ,A(2  )),(PSI   ,A(3  )),
     1            (PHIR  ,A(4  )),(THETR ,A(5  )),(PSIR  ,A(6  )),
     2            (PHID  ,A(7  )),(THEU  ,A(8  )),(PSIO  ,A(9  )),
     3            (SPHI  ,A(10 )),(CPHI  ,A(11 )),(STHT  ,A(12 )),
     4            (CTHT  ,A(13 )),(SPSI  ,A(14 )),(CPSI  ,A(15 )),
     5            (PH    ,A(37 )),(QA    ,A(38 )),(RH    ,A(39 ))
       EQUIVALENCE(PSIS,B(84))
C
      EQUIVALENCE (PLB   ,A(43 )),(QLB   ,A(44 )),(RLB   ,A(45 )),
     1            (PT    ,A(46 )),(QT    ,A(47 )),(RT    ,A(48 )),
     2            (PHWN  ,A(49 )),(QBWN  ,A(50 )),(RBWN  ,A(51 )),
     3            (PTURB ,A(52 )),(QTURB ,A(53 )),(RTURB ,A(54 )),
     4            (PBD   ,A(55 )),(QBD   ,A(56 )),(RBD   ,A(57 )),
     5            (XMC(1),A(120)),(TTL   ,A(164)),(TTM   ,A(165)),
     6            (TTN   ,A(166))
C
      EQUIVALENCE (TIME,A(303))
      EQUIVALENCE (A(358),D2R),(A(359),R2D)
      EQUIVALENCE (A(0375),XMCC1 )
      EQUIVALENCE (A(0376),XMCC2 )
      EQUIVALENCE (A(0377),XMCC3 )
      EQUIVALENCE (A(0378),XMCC4 )
      EQUIVALENCE (A(0379),XMCC5 )
      EQUIVALENCE (A(0380),XMCC6 )
      EQUIVALENCE (A(0381),XMCC7 )
C
      EQUIVALENCE (IMODE,IA(1))
      EQUIVALENCE(IA(16),I2DOF)
C
      DOUBLE PRECISION DTIME
      DATA TS/0.0/
      DATA I2DOF/0/
       DATA PI/3.1415927/
C
C
```

```
C*****RROTATE CALCULATES PBD,QBD,RBD,INTEGRATES TO GET PB,QB,RB,CALCULATES
C
C*****PBWN,QBWN,RBWN,PT,QT,RT,THED,PSID,PHID,INTEGRATES TO GET THEIR,
C
C*****PSIR,PHIR CALCULATES THET,PSI,PHI
C
C
C*****INPUTS- TTL,TTM,TTN,XMC(1) THROUGH XMC(10),PLB,QLB,RLB,PTURB,QTURB,
C
C*****      RTURB,SPHI,CPHI,STHT,CTHT
C
C
C*****OUTPUTS-PBD,QBD,RBD,PB,QB,RB,PBWN,QBWN,RBWN,PT,QT,RT,THED,PSID,PHI
C
C*****           THETR,PSIR
C
C
      IF(IMODE.EQ.-1) GO TO 1
      GO TO 2
00001 PSISP=PSIS
      PSIRP=PSIR
00002 CONTINUE
     PBD=(XMC(1)*RB+XMC(2)*PB)*QB+XMC(3)*TTL+XMC(4)*TTN
    1            + XMCC1*QB + XMCC5*RB - XMCC6*PB
     QBD=XMC(5)*RB*PB+XMC(6)*(RB**2-PB**2)+XMC(7)*TTM
    1            - XMCC2*RB + XMCC3*PB
     RBD=(XMC(8)*PB+XMC(9)*RB)*QB+XMC(4)*TTL+XMC(10)*TTN
    1'           + XMCC4*QB + XMCC6*RB - XMCC7*PB
      IF(IMODE) 10,30,20
00010 QBDP=QBD
      RBDP=RBD
      PBDP=PBD
      DTO2=.5*DELT
      DTIME=0.
      TIME=0.
      GO TO 30
 20   PB=PB+DTO2*(3.0*PBD-PBDP)
      PBDP=PBD
      QB=QB + DTO2*(3.0*QBD-QBDP)
      QBDP=QBD
      TIME1=TIME+DELT
       IF(IZDOF.EQ.1) QB=.017452/DELT*(TAB1(TIME1,0.,0.,67,1)-TAB1(
    1  TIME,0.,0.,67,1))
      RB=RB + DTO2*(3.0*RBD-RBDP)
      RBDP=RBD
      DTIME=DTIME+DELT
      TIME=DTIME
00030 PT=PB-PLB
      QT=QB-QLB
      RT=RB-RLB
      PBWN=PB+PTURB
      QBWN=QB+QTURB
      RBWN=RB+RTURB
      THED=QT*CPHI-RT*SPHI
       CTHT1=CTHT
```

```
      IF(ABS(CTHT1).LT..001) CTHT1=SIGN(.001,CTHT)
      PSID=(QT*SPHI+RT*CPHI)/CTHT1
      PHID=PT+PSID*STHT
      IF(IMODE) 60,70,50
00050 THETR=THETR + DTO2*(THED+DUM)
      IF(I2DOF.EQ.1) THETR=.017452*TABl(TIME,0.,0.,67,1)
      PSIR=PSIR + DTO2*(PSID+DUMB)
      PHIR=PHIR + DTO2*(PHID+DUMA)
C      HEADING CLAMP WHEN GEARS TOUCH
       IF(ITOUCH(1).EQ.0) PSIRP=PSIR
       IF(ITOUCH(1).EQ.1.AND.ITOUCH(2).EQ.1) PSIR=PSIRP+PSIS-PSISP
        PSIRP=PSIR
       PSISP=PSIS
C      REDUCE ANGLES TO +/- 180 DEGREES
       I1=THETR/PI
      IF(I1.EQ.0) GO TO 51
      THETR=-(I1+1)*PI+THETR
 51   I2=PSIS/PI
      IF(I2.EQ.0) GO TO 52
      PSIR=-(I2+1)*PI+PSIR
 52   IF(I3.EQ.0) GO TO 53
      PHIR=PHIR-(I3+1)*PI
 53   CONTINUE
00060 THET=THETR*R2D
      PHI=PHIR*R2D
      PSI=PSIR*R2D
      DUM=THED
      DUMB=PSID
      DUMA=PHID
00070 CONTINUE
      RETURN
      END
```

****

```
      SUBROUTINE ARDC62(ALT,SPSND,DENSTY)
C
C     AUTHOR M P RUHWEN AND R E MCFARLAND FOR 6400
C     SIGMA SEVEN VERSION BY R SHIRLEY
C     IBM 360 VERSION BY D JONES -CSC- FEB 73
C
C
C     CALLING SEQUENCE IS
C
C
C               CALL ARDC62(ALT,SPSND,DENSTY)
C
C
C
      DIMENSION SPS(121),DENS(121)
      DIMENSION DENS1(20),DENS2(20),DENS3(20),DENS4(20)
     1,         DENS5(20),DENS6(20),SPS1(20),SPS2(20),SPS3(20)
     2,         SPS4(20),SPS5(20),SPS6(20)
      EQUIVALENCE (DENS(1),DENS1(1)),(DENS(21),DENS2(1))
```

```
     1.          (DENS(41),DENS3(1)),(DENS(61),DENS4(1))
     2,          (DENS(81),DENS5(1)),(DENS(101),DENS6(1))
     3,          (SPS(1),SPS1(1)),(SPS(21),SPS2(1)),(SPS(41),SPS3(1))
     4,          (SPS(61),SPS4(1)),(SPS(81),SPS5(1)),(SPS(101),SPS6(1))
C
      DATA DENS1/2.37701E-3
     1.                  2.24098E-3
     1.                  2.11099E-3
     1.                  1.98684E-3
     1.                  1.86836E-3
     1.                  1.75537E-3
     1.                  1.64766E-3
     1.                  1.54511E-3
     1.                  1.44751E-3
     1.                  1.35469E-3
     1.                  1.26649E-3
     1.                  1.18276E-3
     1.                  1.10333E-3
     1.                  1.02805E-3
     1.                  .95676E-3
     1.                  .88432E-3
     1.                  .82557E-3
     1.                  .76538E-3
     1.                  .70860E-3
     1.                  .64419E-3/
      DATA DENS2/.585146E-3
     1.                  .531517E-3
     1.                  .482801E-3
     1.                  .438554E-3
     1.                  .398359E-3
     1.                  .361850E-3
     1.                  .328688E-3
     1.                  .298561E-3
     1.                  .271197E-3
     1.                  .246341E-3
     1.                  .223765E-3
     1.                  .203256E-3
     1.                  .184627E-3
     1.                  .167616E-3
     1.                  .151855E-3
     1.                  .137615E-3
     1.                  .124744E-3
     1.                  .113107E-3
     1.                  .102584E-3
     1.                  .093066E-3/
      DATA DENS3/.084453E-3
     1.                  7.6714E-5
     1.                  6.9701E-5
     1.                  6.3349E-5
     1.                  5.7588E-5
     1.                  5.2370E-5
     1.                  4.7635E-5
     1.                  4.3341E-5
     1.                  3.9443E-5
     1.                  3.5908E-5
```

```
1,                    3.2696E-5
1,                    2.9781E-5
1,                    2.7132E-5
1,                    2.4698E-5
1,                    2.2414E-5
1,                    2.0357E-5
1,                    1.8501E-5
1,                    1.6827E-5
1,                    1.5315E-5
1,                    1.3948E-5/
 DATA DENS4/1.271E-5
1,                    1.1592E-5
1,                    1.0579E-5
1,                    9.6601E-6
1,                    8.8271E-6
1,                    8.0707E-6
1,                    7.3838E-6
1,                    6.7594E-6
1,                    6.1916E-6
1,                    5.6744E-6
1,                    5.2045E-6
1,                    4.7757E-6
1,                    4.3847E-6
1,                    4.0282E-6
1,                    3.7026E-6
1,                    3.4052E-6
1,                    3.1333E-6
1,                    2.8848E-6
1,                    2.6627E-6
1,                    2.4683E-6/
 DATA DENS5/2.2881E-6
1,                    2.1212E-6
1,                    1.9664E-6
1,                    1.8230E-6
1,                    1.6900E-6
1,                    1.5668E-6
1,                    1.4525E-6
1,                    1.3524E-6
1,                    1.2588E-6
1,                    1.1713E-6
1,                    1.0896E-6
1,                    1.0132E-6
1,                    9.4195E-7
1,                    8.7537E-7
1,                    8.1326E-7
1,                    7.5532E-7
1,                    7.0126E-7
1,                    6.5085E-7
1,                    6.0367E-7
1,                    5.6011E-7/
 DATA DENS6/5.1932E-7
1,                    4.8134E-7
1,                    4.4746E-7
1,                    4.1664E-7
1,                    3.8732E-7
```

```
1,                3.5979E-7
1,                3.3397E-7
1,                3.0976E-7
1.                2.8712E-7
1,                2.6592E-7
1,                2.4609E-7
1,                2.2757E-7
1.                2.1027E-7
1.                1.9412E-7
1,                1.7906E-7
1,                1.6503E-7
1,                1.5197E-7
1,                1.3981E-7
1.                1.2851E-7
1,                1.1802E-7/
 DATA DENS(121)/1.0827E-7/
 DATA SPS1/1116.424
1.                1108.722
1,                1100.965
1.                1093.154
1,                1085.286
1.                1077.360
1,                1069.376
1.                1061.333
1.                1053.227
1.                1045.059
1.                1036.826
1,                1028.527
1,                1020.161
1,                1011.726
1,                1003.220
1.                994.641
1,                985.968
1,                977.256
1.                968.448
1,                968.053 /
 DATA SPS2/968.053
1,                968.053
1.                968.053
1,                968.053
1.                968.053
1,                968.053
1,                968.053
1,                968.053
1,                968.053
1.                968.053
1.                968.053
1,                968.053
1,                968.053
1,                968.314
1,                968.676
1,                971.034
1,                972.390
1,                973.745
1,                975.049
```

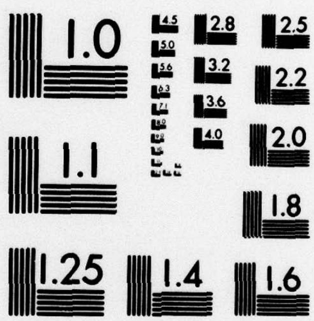
```
1.                           976.450 /
 DATA SPS3/977.799
1.                           978.95
1.                           980.29
1.                           981.62
1.                           982.95
1.                           984.28
1.                           985.61
1.                           986.93
1.                           988.26
1.                           989.58
1.                           990.90
1.                           992.21
1.                           993.53
1.                           995.41
1.                           999.07
1.                           1002.72
1.                           1006.36
1.                           1009.98
1.                           1013.59
1.                           1017.19 /
 DATA SPS4/1020.77
1.                           1024.34
1.                           1027.89
1.                           1031.44
1.                           1034.97
1.                           1038.48
1.                           1041.99
1.                           1045.48
1.                           1048.96
1.                           1052.43
1.                           1055.88
1.                           1059.33
1.                           1062.76
1.                           1066.18
1.                           1069.59
1.                           1072.99
1.                           1076.37
1.                           1079.75
1.                           1082.02
1.                           1082.02 /
 DATA SPS5/1082.02
1.                           1082.02
1.                           1082.02
1.                           1082.02
1.                           1082.02
1.                           1082.02
1.                           1082.02
1.                           1079.63
1.                           1077.23
1.                           1074.82
1.                           1072.40
1.                           1069.98
1.                           1067.56
1.                           1065.13
```

```
      1,              1062.69
      1,              1060.25
      1,              1057.81
      1,              1055.36
      1,              1052.90
      1,              1050.44 /
     DATA SPS6/1047.98
      1,              1045.50
      1,              1040.63
      1,              1035.65
      1,              1030.65
      1,              1025.62
      1,              1020.57
      1,              1015.49
      1,              1010.39
      1,              1005.26
      1,              1000.11
      1,              994.94
      1,              989.73
      1,              984.50
      1,              979.25
      1,              973.96
      1,              968.65
      1,              963.31
      1,              957.94
      1,              952.55 /
     DATA SPS(121)/947.12/
C
C
     H=ALT
     IF(ALT.LT.0.0) H=0.0
     IF(ALT.GT.240000.) H=240000.
C
C  FIND ALT INCREMENT AND FRACTION  N
C
     IALT=H*0.0005  +1
     ALTF=(H-(IALT-1)*2000)*0.0005
C
C   CALCULATE DENSITY AND SPEED OF SOUND
     DENSTY=DENS(IALT)-(DENS(IALT)-DENS(IALT+1))*ALTF
     SPSND=SPS(IALT)-(SPS(IALT)-SPS(IALT+1))*ALTF
C
C
     RETURN
     END

*****************************************************************************

     SUBROUTINE BALFBET(DELT)
C
C    *************************************************************
C    *                                                           *
C    * BALFBET CALCULATES ALFD,BETD,ALFAR,BETAR,ALFA,BETA, AND SINE AND *
```

```
C        AND UBD,VBD,WBD,AND VRW--AS OF APRIL 20,1974 (FROM BATMOSPH)
C     *                                                                  *
C     * COSINE OF ALFAR AND BETAR                                        *
C     *                                                                  *
C     *                                                                  *
C     * INPUTS- UB,VB,WB,                                                *
C     *                                                                  *
C     * OUTPUTS-ALFAR,ALFA,SALPH,CALPH,BETAR,BETA,ALFD,BETD,SBETA,CBETA   *
C     *                                                                  *
C     ********************************************************************
C     * UPDATED 27 DEC 73 TO REFLECT CHANGES MADE TO CURRENT FSAA VERSION*
C     ********************************************************************
C
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
C
      EQUIVALENCE (ALFA  ,A(25 )),(BETA  ,A(26 )),(ALFAR ,A(27 )),
     1            (BETAR ,A(28 )),(ALFD  ,A(29 )),(BETD  ,A(30 )),
     2            (SALPH ,A(31 )),(CALPH ,A(32 )),(SBETA ,A(33 )),
     3            (CBETA ,A(34 )),(QB    ,A(38 )),(RB    ,A(39 )),
     4            (UB    ,A(58 )),(VB    ,A(59 )),(WB    ,A(60 )),
     5            (D2R   ,A(358)),(R2D   ,A(359))
C
      EQUIVALENCE (A(413),UBD),(A(414),VBD),(A(415),WBD)
     1,(A(70),VRW),(A(416),VTWN),(A(417),VTWE)
     2,(A(418),VTWD),(A(88),VND),(A(89),VED),(A(90),VDD)
     3,(A(46),PT),(A(47),QT),(A(48),RT),(A(16),T11),(A(17),T21)
     4 ,(A(18),T31),(A(19),T12),(A(20),T22),(A(21),T32)
     5,(A(22),T13),(A(23),T23),(A(24),T33)
      EQUIVALENCE (ITOMTR, IA(187))
      EQUIVALENCE (IMODE,IA(1))
C
      DATA ITOMTR/0/
C
C     ********************************************************************
C     * "ALFA" IS ANGLE OF ATTACK IN DEGREES                             *
C     * "BETA" IS ANGLE OF SIDESLIP IN DEGREES                           *
C     *                                                                  *
C     * NOTATION AS FOLLOWS... R-SUFFIX INDICATES RADIAN MEASURE         *
C     *                        D-SUFFIX INDICATES DERIVATIVE             *
C     *                        S-PREFIX INDICATES SINE OF ANGLE          *
C     *                        C-PREFIX INDICATES COSINE OF ANGLE        *
C     ********************************************************************
C
C
C
C     ********************************************************************
C     * TO PROTECT ALFA AND BETA FOR UB=VB=WB=0.0                        *
C     ********************************************************************
C
      IF(ABS(UB).LT.0.00001) UB=SIGN(0.00001,UB)
C
      ALFAR=ATAN2(WB,UB)
      ALFA=ALFAR*R2D
      SALPH=SIN(ALFAR)
      CALPH=COS(ALFAR)
```

AD-A073 587 NAVAL AIR DEVELOPMENT CENTER WARMINSTER PA AIR VEHICL--ETC F/G 1/2
A COMPUTERIZED VSTOL/SMALL PLATFORM LANDING DYNAMICS INVESTIGAT--ETC(U)
SEP 77 R L NAVE
UNCLASSIFIED NADC-77024-30 NL

2 OF 4
AD
A073587



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
C
C
C     ***************************************************************
C     *                                                             *
C     * BTRANSFORM CALCULATES SINES AND COSINES OF PSIR,THETR,PHIR, AND *
C     *                                                             *
C     * TRANSFORMATION ELEMENTS T11,T21,T31,T12,T22,T32,T13,T23,T33  *
C     *                                                             *
C     *                                                             *
C     * INPUTS-PSIR,THETR,PHIR                                      *
C     *                                                             *
C     * OUTPUTS-SPSI,STHT,SPHI,CPSI,CTHT,CPHI,T11,T21,T31,T12,T22,T32, *
C     *       T13,T23,T33                                           *
C     *                                                             *
C     ***************************************************************
C
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
C
      EQUIVALENCE (PHIR   ,A(4   )),(THETR  ,A(5   )),(PSIR   ,A(6   )),
     1            (SPHI   ,A(10  )),(CPHI   ,A(11  )),(STHT   ,A(12  )),
     2            (CTHT   ,A(13  )),(SPSI   ,A(14  )),(CPSI   ,A(15  )),
     3            (T11    ,A(16  )),(T21    ,A(17  )),(T31    ,A(18  )),
     4            (T12    ,A(19  )),(T22    ,A(20  )),(T32    ,A(21  )),
     5            (T13    ,A(22  )),(T23    ,A(23  )),(T33    ,A(24  ))
C
C
C
      SPHI=SIN(PHIR)
      CPHI=COS(PHIR)
      SPSI=SIN(PSIR)
      CPSI=COS(PSIR)
      STHT=SIN(THETR)
      CTHT=COS(THETR)
C
      T11=CTHT*CPSI
      T21=SPHI*STHT*CPSI-CPHI*SPSI
      T31=CPHI*STHT*CPSI+SPHI*SPSI
      T12=CTHT*SPSI
      T22=SPHI*STHT*SPSI+CPHI*CPSI
      T32=CPHI*STHT*SPSI-SPHI*CPSI
      T13=-STHT
      T23=SPHI*CTHT
      T33=CPHI*CTHT
C
      RETURN
C
      END

*************************************************************************

      SUBROUTINE BVELOC
C
C     ***************************************************************
```

```
      DUM2=UB**2+WB**2
      DUM = SQRT(DUM2)
      DUM1      =  SIGN(DUM,UB)
      BETAR     =  ATAN2(VB,DUM1)
      BETA=BETAR*R2D
      SBETA=SIN(BETAR)
      CBETA=COS(BETAR)
C VRW CALCULATION (FROM BATMOSPH)
      VRW2=DUM2+VB**2
      VRW=SQRT(VRW2)
       IF(IMODE.EQ.-1) GO TO 35
C
C WIND ACCELERATION INCLUDING TURBULENCE AND WIND VARIATIONS
C
       IF(DELT.EQ.0.) GO TO 35                                    99
      ATWN=(VTWN-VTWNP)/DELT
      ATWE=(VTWE-VTWEP)/DELT
       ATWD=(VTWD-VTWDP)/DELT
       GO TO 36
00035  ATWN=0.
       ATWE=0.
       ATWD=0.
00036  CONTINUE
      VTWNP=VTWN
      VTWEP=VTWE
      VTWDP=VTWD
C RELATIVE ACCELERATIONS
      RAN=VND-ATWN
      RAE=VED-ATWE
      RAD=VDD-ATWD
C BODY AXIS ACCELERATIONS
      UBD=RT*VB-QT*WB+T11*RAN+T12*RAE+T13*RAD
      VBD=PT*WB-RT*UB+T21*RAN+T22*RAE+T23*RAD
      WBD=QT*UB-PT*VB+T31*RAN+T32*RAE+T33*RAD
C
       IF(IMODE) 38,39,39
00038  IF(ITOMTR.NE.1) GO TO 39
       ALFD=0.
       BETD=0.
       GO TO 40
00039  IF(VRW-.1)41,41,42
00041  ALFD=0.
       BETD=0.
       GO TO 40
00042  ALFD=(UB*WBD-WB*UBD)/DUM2
      BETD=(DUM2*VBD-(UB*UBD+WB*WBD)*VB)/(DUM1*VRW2)
00040 RETURN
      END
```

****

```
      SUBROUTINE BTRANS
```

```
C    *                                                                   *
C    * BVELOCITY CALCULATES THE INERTIAL VELOCITIES WITH RESPECT TO THE  *
C    *                                                                   *
C    * INERTIAL WIND COMPONENTS, AND THE BODY AXIS VELOCITIES U,V,W      *
C    *                                                                   *
C    *                                                                   *
C    * INPUTS- VN,VFE,VD,VNW,VEW,VDW,T11,T21,T31,T12,T22,T32,T13,T23,T33*
C    *                                                                   *
C    * OUTPUTS-VNR,VER,VDR,UB,VB,WB                                      *
C    *        UTURB,VTURB,WTURB                                          *
C    *                                                                   *
C    *********************************************************************
C
C
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
C
      EQUIVALENCE (T11    ,A(16 )),(T21    ,A(17 )),(T31    ,A(18 )),
     1            (T12    ,A(19 )),(T22    ,A(20 )),(T32    ,A(21 )),
     2            (T13    ,A(22 )),(T23    ,A(23 )),(T33    ,A(24 )),
     3            (UB     ,A(58 )),(VB     ,A(59 )),(WB     ,A(60 )),
     4            (UTURB  ,A(61 )),(VTURB  ,A(62 )),(WTURB  ,A(63 )),
     5            (VN     ,A(64 )),(VD     ,A(66 )),(VEE    ,A(67 )),
     6            (VNR    ,A(72 )),(VER    ,A(73 )),(VDR    ,A(74 )),
     7            (VNW    ,A(76 )),(VEW    ,A(77 )),(VDW    ,A(78 ))
      EQUIVALENCE (IA(185),IETURB),(A(419),VNTURB)
     1,(A(420),VETURB),(A(421),VDTURB),(A(416),VTWN)
     2,(A(417),VTWE),(A(418),VTWD)
C
C
C
C
C                 ****************************************************
C                 * NORTH, EAST, DOWN VELOCITIES INCLUDING WINDS     *
C                 * W.R.T. THE RUNWAY                                *
C                 ****************************************************
C IF IETURB .EQ. 1, THE RANDOM TURBELENCE IS ALREADY GENERATED IN THE
C EARTH AXES AS VNTURB, VETURB, VDTURB.  OTHERWISE, THIS ROUTINE
C EXPECTS UTURB, VTURB, WTURB.
C
C
C                 ****************************************************
C                 *  BODY AXIS VELOCITIES INCLUDING TURBULENCE       *
C                 ****************************************************
      IF (IETURB .EQ. 0) GO TO 10
      UTURB = T11*VNTURB+T12*VETURB+T13*VDTURB
      VTURB = T21*VNTURB+T22*VETURB+T23*VDTURB
      WTURB = T31*VNTURB+T32*VETURB+T33*VDTURB
      GO TO 20
00010 VNTURB= T11*UTURB+T21*VTURB+T31*WTURB
      VETURB = T12*UTURB+T22*VTURB+T32*WTURB
      VDTURB = T13*UTURB+T23*VTURB+T33*WTURB
C
00020 VTWN = VNW+VNTURB
      VTWE = VEW+VETURB
      VTWD = VDW+VDTURB
```

```
C
      VNR = VN-VTWN
      VER = VEE-VTWE
      VDR = VD-VTWD
      UR = T11*VNR+T12*VER+T13*VDR
      VR = T21*VNR+T22*VER+T23*VDR
      WR = T31*VNR+T32*VER+T33*VDR
C
C
C
      RETURN
      END
```

```
****************************************************************************************

      SUBROUTINE TABRD(NUMTBL,NZ,NG)
      COMMON/TABLE/NUMPTS(3500)
      COMMON/TABOUT/NIMTBL,ISOO
      COMMON/XTRAP/METH,IACT(3,93)
      COMMON/FIND/IPREV(3,93)
      DIMENSION ISCR(3)
      DIMENSION XUMPTS(3500),LABEL(25)
      EQUIVALENCE (XUMPTS(1),NUMPTS(1)),(DUMMY(1),MUMMY)
      DIMENSION DUMMY(10)
      DATA ICNT1/0/
      MCR=0
      ICNT1=ICNT1+1
      PRINT 500,ICNT1
  500 FORMAT(1H0,10X,30H SUBROUTINE TABRD ENTERED        ,10X,I3,
     115X,6HTIMES  )
      MCR=1
      GO TO 10
      ENTRY TABWRT
      PRINT 100
  100 FORMAT(1H1)
      MCR=1
   10 IZ=IABS(NZ)
      NIMTBL = NUMTBL
      NG=0
      NUMPTS(1)=IZ+IZ+IZ
      ITALE=0
  102 READ(97,57) K,L1N,L2N,ISCR,LABEL,ISEO
      IF(K.EQ.0) GOTO 4
      ITBLE=ITBLE+1
      DO 20 ICHSE=1,3
      IACT(ICHSE,ITBLE)=ISCR(ICHSE)+1
20    IPREV(ICHSE,ITBLE)=1
      IPREV(3,ITBLE)=ITBLE
      IF(MCR.EQ.0) GOTO 3
    4 PRINT  1,   K,L1N,L2N,LABEL,ISEQ
    1 FORMAT(3I5,14X,25A2,I46)
   57 FORMAT(8X,I4,2I2,3I1,1X,25A2,I2)
    3 IF(ISEQ) 69,58,69
```

```
 58  IF(K) 99, 99, 59
 59  M = IZ + NIMTBL
     NUMPTS(M) = L1N
     M = M + IZ
     NUMPTS(M) = L2N
     IF(NUMTBL-NIMTBL)17,70,17
 17  NUMPTS(NIMTBL) = MUMMY
 70  N1 = (L1N-1) / 9 + 1
     DO 66 IS = 1,N1
     L3 = (IS-1) * 9 + 1
     IF (IS-N1) 60, 61, 60
 60  L4 = L3 + 8
     GO TO 62
 61  L4 = L1N
 62  L5 = NUMPTS(NIMTBL) + 1
     L6 = L5 + L3
     L7 = L5 + L4
     JJ = 0
     LM = L5 + L1N
     LN = LM + L2N
 63  READ(97,64) (DUMMY(K),K=1,10),ISEQ
 64  FORMAT (10E7.0,I2)
     IF(MCR.EQ.0) GO TO 5
  6  PRINT  2,   DUMMY,ISEQ
  2  FORMAT(10E12.4,I5)
  5  XNUMPTS(L5)= DUMMY(1)
     K = 2
     DO 65 J = L6,L7
     XNUMPTS(J) = DUMMY(K)
     NMAX=J
 65  K = K+1
     ISOQ=(IS-1)*(L2N+1)+JJ+1
     IF(ISEQ-ISOQ) 69,66,69
 66  L6 = LN + L3
     L7 = LN + L4
     L5 = LM + 1 + JJ
     IF (JJ-L2N) 67,68,69
 67  JJ = JJ + 1
     LN = LN + L1N
     GO TO 63
 68  CONTINUE
109  MUMMY = NUMPTS(NIMTBL) + (L1N+1) * (L2N+1)
108  NIMTBL = NIMTBL + 1
     GO TO 102
 69  NG = 1
 90  PRINT 110,NMAX
110  FORMAT(* ...*,I6,2X,* LOCATIONS IN NUMPTS ARRAY WERE LOADED *)
     RETURN
     END
```

****

```
     FUNCTION TAB1(A,B,C,N,ND)
```

```
      DATA IERPRT/1/
      CALL LOOKUP(A,B,C,N,N+ND-1,FC,NG)
      NG1=NG+1
      GOTO (6,2,2,7,2,2,7,7),NG1
 7    IF(IERPRT.EQ.1) GO TO 6
      PRINT 1000,N,A,B,C,FC
    6 TAB1=FC
      RETURN
    2 PRINT  1001,NG
      RETURN
1000  FORMAT(* ..... OFF TABLE *,I3,* ARGS=*,3(E10.4,1X),* .. FUNC.= *,
     1     E10.4)
1001  FORMAT(* ...... TABLE ERROR , NG = *,I1)
      END

****************************************************************************************

      SUBROUTINE LOOKUP(A1,A2,A3,MINTBL,MAXTBL,FCT,NG)
C     **************************************************************
C       METH:  KEY FOR TYPE SEARCH
C            1 - STANDARD ORDER SEARCH
C            2 - MEMORY SEARCH
C            3 - BINARY SEARCH
C        IACT(ARG,TABLE):  KEY FOR OFF TABLE ACTION
C            1 - SET FCT=0,NG=3
C            2 - HOLD LAST TABLE VALUE OF FCT, SET NG=6
C            3 - LINEARLY EXTRAPOLATE FCT, SET NG=7
C     **************************************************************
      COMMON/TABOUT/NINTBL,ISOO
       COMMON/TABLE/NUMPTS(3500)
      COMMON/XTRAP/METH,IACT(3,93)
      COMMON/FIND/IPREV(3,93)
      EQUIVALENCE (X(1),NUMPTS(1))
       DIMENSION X (1),ARG(3)
      DATA INIT/0/
       METH=2
      ELOG2=ALOG(2.0)
      FCT=0.0
      NG=0
      I7=NUMPTS(1)/3
00070 IF(MINTBL-MAXTBL)71,71,110
00071 DO 73 II=MINTBL,MAXTBL
      NJ=NUMPTS(II)+1
      IF(A3-X(NJ))72,74,73
00072 IF(II-MINTBL) 110,112,75
00073 CONTINUE
      II=MAXTBL
      GO TO 112
00075 IK= 1
      IL =2
      NM=NJ
00101 DO 97 IF=IK,IL
      NJ=NUMPTS(II)+1
```

```
      NI = IZ+II
      IO =NUMPTS(NI)
      IP =IO+NJ
      IARG=1
      ARG(1)=A1
      NJIP=NJ
      IOIH=IO
      IF(A1.GT.X(NJ+IO)) GO TO 212
      IF(A1  .LT. X(NJ+1))  GO TO 312
      GO TO (40,41,42),METH
00076 IG=IGIH
      IO=IOIA
      NN=IND
      IPREV(1,II)=IO
      GO TO 80
C        BINARY SEARCH SECTION
00042 ILOW=1
      IHI=IOIP
      IF(X(IHI+NJIP) .NE. ARG(IARG)) GO TO 43
      IOIA=IHI
00056 IGIH=-1
      IND=NJIP+IOIA
00057 GO TO (76,81),IARG
43    IF(X(ILOW+NJIP)  .NE. ARG(IARG)) GO TO 44
      IOIA=ILOW
      GO TO 56
44    EL=IOIH
      NUMIT=(ALOG(EL)/ELOG2)+1.0
      DO 45 KNT=1,NUMIT
      MID=ILOW+(IHI-ILOW)/2
      IND=NJIP+MID
      IF(ARG(IARG)-X(IND)) 46,55,48
46    IF(ARG(IARG)-X(IND-1))  47,51,50
47    IHI=MID
      GO TO 45
48    IF(ARG(IARG)-X(IND+1))  52,53,49
49    ILOW=MID
45    CONTINUE
      GO TO 110
55    IOIA=MID
      GO TO 56
51    IOIA=MID-1
      GO TO 56
53    IOIA=MID+1
      GO TO 56
50    IOIA=MID
54    IGIH=1
      IND=NJIP+IOIA
      GO TO 57
52    IOIA=MID+1
      GO TO 54
C        MEMORY SEARCH SECTION
41    IOIA=IPREV(IARG,II)
      IND=NJIP+IOIA
      IF(ARG(IARG)-X(IND))  60,61,62
```

```
60     IF(ARG(IARG)-X(IND-1))  63,64,65
63     IND=IND-1
       GO TO 60
64     IND=IND-1
61     IOIA=IND-NJIP
       GO TO 56
65     IOIA=IND-NJIP
       GO TO 54
62     IF(ARG(IARG)-X(IND+1))  66,67,68
68     IND=IND+1
       GO TO 62
67     IND=IND+1
       GO TO 61
66     IND=IND+1
       GO TO 65
C         STANDARD ORDER SEARCH
40     DO 77 IOIA=1,IOIB
       IND=NJIP+IOIA
       IF(ARG(IARG)-X(IND)) 54,56,77
77     CONTINUE
       GO TO 110
00080  NI=NI+I2
       IA = NUMPTS(NI)
       IARG=2
       ARG(2)=A2
       NJIP=IP
       IOIB=IB
       IF(A2  .GT. X(IP+IB)) GO TO 212
       IF(A2  .LT. X(IP+1))  GO TO 312
       GO TO (40,41,42), METH
81     IM=IGIM
       IA=IOIA
       NS=IND
       IPREV(2,II)=IA
00085  NE=IP+IM+IO+IO*IA-IO
       NR=NE-IO
       IF(IB+IM) 86,86,91
00086  IF (X(NE)-99998.5E9)87,113,113
00087  FCT = X(NE)
       GO TO 95
00088  IF(IB) 89,110,93
   89  IF(AMAX1(X(NE),X(NR))-99998.5E9)90,113,113
   90  FCT =X(NE)-(X(NS)-A2)*(X(NE)-X(NR))/(X(NS)-X(NS-1))
       GO TO 95
   91  IF(AMAX1(X(NE),X(NR),X(NE-1),X(NR-1))-99998.5E9)92,113,113
   92  FCT = ((X(NS)-A2)*((X(NN)-A1)*X(NR-1)-(X(NN-1)-A1)*X(NR)
      1)-(X(NS-1)-A2)*((X(NN)-A1)*X(NE-1)-(X(NN-1)-A1)*X(NE)))
      2/((X(NS)-X(NS-1))*(X(NN)-X(NN-1)))
       GO TO 95
   93  IF(AMAX1(  X(NE),  X(NE-1))-99998.5E9) 94,113,113
   94  FCT =  X(NE)-(  X(NN)-A1)*(  X(NE)-  X(NE-1))/(  X(NN)-  X(NN-1))
   95  GO TO (96,98,99),IF
   96  DIMST6 =FCT
       GO TO 97
C           X OR Y VALUE OFF TOP OF TABLE
```

```
212   IF(IACT(IARG,II) .GT. 1) GO TO 11
      GO TO 12
11    IQIA=IQIB
      IND=NJIP+IQIA
      IGIM=1
      NG=7
      IF(IACT(IARG,II) .EQ. 3) GO TO (76,81), IARG
      IGIM=-1
      NG=6
      GO TO (76,81), IARG
C        X OR Y VALUE OFF BOTTOM OF TABLE
312   IPNTR=IACT(IARG,II)
      GO TO (12,13,14), IPNTR
12    NG=3
      RETURN
13    IGIM=-1
      IQIA=1
      IND=NJIP+IQIA
      NG=6
      GO TO (76,81), IARG
14    IGIM=1
      IQIA=2
      IND=NJIP+IQIA
      NG=7
      GO TO (76,81), IARG
97    II=II-1
98    FCT=DUMSTG-(X(NM)-A3)*(DUMSTG-FCT)/(X(NM)-X(NJ))
99    RETURN
76    IK=3
      IL=3
      GO TO 101
110   NG=2
      GO TO 99
C        Z VALUE OFF  TABLE
112   IPNTR=IACT(3,II)
      GO TO (12,21,22), IPNTR
21    IK=3
      IL=3
      NG=6
      NM=NJ
      GO TO 101
22    IF(MINTBL  .EQ. MAXTBL) GO TO 12
      IF(II .EQ. MINTBL)  II=MINTBL+1
      IK=1
      IL=2
      NM=NJ
      NG=6
      GO TO 101
 113  NG =4
      GO TO 99
      END
```

```
      SUBROUTINE TIMEHIS(TIME)
       COMMON/XFLOAT/A(500)/IFIXED/IA(200)
       COMMON/CBF/B(200)/ICHF/IB(50)
       COMMON/GAIN1/GA(100)
      COMMON/TBLOCK/TPRNTA(500),TPRNTB(200)
      COMMON/DBLOCK/IPLOT(14),IPRNT(11)
      COMMON/CONEM/EMCONV(15),PSCALE(2,14)
      DIMENSION IP1(11),IP2(11),IP3(11),RPRNT(11),DPRNT(11)
      DATA(IPRNT(I),I=1,11)/ 11025,11002,51080,11001,11003,51083,51070,
     1  51174,12001,12019,12017      /
      DATA TIT1/6HTIME-S     /
      IF(TIME.GT.0.) GO TO 10
      DO 1 J=1,11
      IP1(J)=IPRNT(J)/10000
      IP2(J)=(IPRNT(J)-IP1(J)*10000)/1000
      IP3(J)=IPRNT(J)-IP1(J)*10000-IP2(J)*1000
00001 CONTINUE
      DO 2 J=1,11
      IF(IP2(J).EQ.1) RPRNT(J)=TPRNTA(IP3(J))
      IF(IP2(J).EQ.2) RPRNT(J)=TPRNTB(IP3(J))
00002 CONTINUE
      PRINT 3,TIT1, (RPRNT(I),I=1,11)
00003 FORMAT(1X,A6,11(A10,1X))
00010 CONTINUE
      DO 11  J=1,11
      IF(IP2(J).EQ.1) DPRNT(J)=A(IP3(J))*EMCONV(IP1(J))
      IF(IP2(J).EQ.2)DPRNT(J)=B(IP3(J))*EMCONV(IP1(J))
00011 CONTINUE
      PRINT 5,TIME, (DPRNT(I),I=1,11)
00005 FORMAT (1X,F5.2,11(1X,G10.4))
       RETURN
      END
```

****

```
      SUBROUTINE DPLOT
       COMMON/XFLOAT/A(500)/IFIXED/IA(200)
       COMMON/CHF/B(200)/ICHF/IB(50)
       COMMON/GAIN1/GA(100)
      COMMON/TBLOCK/TPRNTA(500),TPRNTB(200)
       COMMON/DAT/DATTIT(15),DATHD(40),CHTSP,SCALE(2,14),
     1  TMMAX,NPLOT(14),ISCALE
       COMMON/DAT2/DATPLT(300,15)
      COMMON/DBLOCK/IPLOT(14),IPRNT(11)
      COMMON/CONEM/EMCONV(15),PSCALE(2,14)
      COMMON/SCP/MTRANS(14),MPLOT(14),IDAT(14)
       DIMENSION DATHD1(6),DATHD2(6),DATHD3(6),DATHD4(6),DATHD5(6),
     1  DATHD6(6),DATHD7(7)
       DIMENSION TBUF(1024)
       EQUIVALENCE(FAC,GA(1))
       EQUIVALENCE(NCNT,IB(36))
       EQUIVALENCE(NHUN,IA(105))
      EQUIVALENCE(WAITIC,A(242))
```

```
      EQUIVALENCE(WAIT,A(177))
      EQUIVALENCE(CG,A(256))
      EQUIVALENCE(VEQIC,A(238))
     EQUIVALENCE(ILTURB,IB(21)),(IWIND,IB(20)),(NCHK,IB(22))
      EQUIVALENCE(RKQ,B(160)),(TLEADP,B(161)),(    RKDRRB,B(167))
      EQUIVALENCE(TLAGRB,B(168)),(RKDRAY,B(169)),(TLEADAY,B(170))
      EQUIVALENCE(TLAGAY,B(171)),(RKDRDA,B(176)),(RPBDA,B(179))
      EQUIVALENCE(TLEADDA,B(180)),(TLAGDA,B(181))
      EQUIVALENCE(TAUACDA,B(186)),(TAUACHS,B(187))
      EQUIVALENCE(IPDAMP,IB(26)),(IQUAMP,IB(27)),(IRDAMP,IB(28))
      EQUIVALENCE(HMAG,B(104)),(PITCHM,B(110))
      DATA (NPLOT (I),I=1,14)/14*1/
      DATA CHTSP/.1968/
      DATA FAC/1./
     DATA((SCALE(I,J),I=1,2),J=1,14)/2.,-2.,2.,-2.,100.,0.,150.,0.,
    1 110.,90.,20.,-5.,20.,-20.,0.,-100.,150.,0.,75.,25.,5.,-5.,10.,
    2-10.,20.,-5.,50.,-50./
      DATA(DATTIT(J),J=1,15)/10HRLONSTK-IN ,10HRLATSTK-IN  ,
    1 10HRUDPED-IN  ,10HTHTNC-D.   ,10HTHROT-D.   ,10HTHET-D.    ,
    2  10HPHI-D.     ,10HPSI-D.      ,10HVRW-F/S      ,10HALT-FT.      ,
    3  10HALTD-F/S   ,10HHE-FT.      ,10HALFA-DEG     ,10HYE-FT.       ,
    4  10HTIME-SEC.      /
     DATA(DATHD(J),J=1,40)/10HAV-8A       ,10HSIMULATION  ,
    110H             ,10H                ,10HRUN=          ,10H          ,
    210HWEIGHT=     ,10H             ,10H  CG=          ,10H         ,
    4  10HVEQIC=       ,10H               ,10HRKQ=              ,10H          ,
    510HLEADP=       ,10H               ,10HRKDRRP=      ,10H          ,
    510HTLAGRP=      ,10H               ,10HRKDRAY=      ,10H         ,
    610HRPBDA=       ,10H               ,10HTLAGRA=      ,10H         ,
    7 10HILTURB=      ,
    7 10H            ,10HIWIND=       ,10H              ,10HNCHK=        ,
    810H             ,10HIPDAMP=      ,10H              ,10HHMAG=        ,
    910H             ,10HPITCHM=      ,10H              ,10HIQDAMP=      ,
    110H            /
      DATA DATHD7(5)/10HCHART SPD    /,DATHD7(7)/10HINCH/SEC     /
      DATA ISCALE/1/
      DATA(EMCONV(I),I=1,15)/1.,4.4418,.45359,14.5919,.3048,.092903,
    1 .0283168,515.3089,47.8115,1.3538,6.4516,2.54,1.,1.,1./
C       ENGLISH TO METRIC CONVERSION EMCONV(I)
C     1          1.                          ------------
C     2          4.4418                      LBF TO NEWTON
C     3          .45359                      LB-M TO KILOGRAMS
C     4          14.5919                     SLUG TO KILOGRAMS
C     5          .3048                       FEET TO METERS
C     6          .092903                     FT**2 TO METER**2
C     7          .0283168                    FT**3  TO METER**3
C     8          515.3089                    SLUG/FT**3  TO KILOGRAM/METER**3
C     9          47.8115                     LB-F/FT**2  TO NEWTON/METER**2
C    10          1.3538                      SLUG-FT**2 TO KILOGRAM-M**2,FT-LB-F TO NTN-M.
C    11          6.4516                      IN**2  TO CM**2
C    12          2.54                        IN TO CM
        PRINT 105
00105   FORMAT(10X,15HDPLOT CALLED        )
      DATHD(3)=DATE(D)
      DATHD(4)=TIME(A)
```

```
00100   FORMAT(I10)
00101   FORMAT(A10)
00102   FORMAT(F10.4)
      REWIND 2
       WRITE(2,100) NRUN
      WRITE(2,102) WAITIC
       WRITE(2,102)CG
       WRITE(2,102) VEQIC
       WRITE(2,102)RKQ
       WRITE(2,102)TLEADP
       WRITE(2,102)RKDRRB
       WRITE(2,102)TLAGRB
       WRITE(2,102)RKDRAY
       WRITE(2,102)RPRDA
       WRITE(2,102)TLAGRB
       WRITE(2,100)   ILTURB
       WRITE(2,100) IWIND
       WRITE(2,100) NCHK
       WRITE(2,100)IPDAMP
       WRITE(2,102)HMAG
       WRITE(2,102)PITCHM
       WRITE(2,100)IQDAMP
       WRITE(2,102) CHTSP
       REWIND 2
       READ(2,101) DATHD(6)
       READ(2,101) DATHD(8)
       READ(2,101) DATHD(10)
       READ(2,101) DATHD(12)
       READ(2,101) DATHD(14)
       READ(2,101) DATHD(16)
       READ(2,101) DATHD(18)
       READ(2,101) DATHD(20)
       READ(2,101) DATHD(22)
       READ(2,101) DATHD(24)
       READ(2,101) DATHD(26)
       READ(2,101) DATHD(28)
       READ(2,101) DATHD(30)
       READ(2,101) DATHD(32)
       READ(2,101) DATHD(34)
       READ(2,101) DATHD(36)
       READ(2,101) DATHD(38)
       READ(2,101) DATHD(40)
       READ(2,101) DATHD7(6)
       XMMAX=TMMAX*CHTSP
      IF(ISCALE.EQ.0) GO TO 400
      DO 200 I=1,14
      IF(ABS(PSCALE(2,I)).GT.ABS(PSCALE(1,I)))PSCALE(1,I)=
     1  ABS(PSCALE(2,I))
00200  CONTINUE
      DO 300 I=1,14
      IF(ABS(PSCALE(1,I)).LT..00001)PSCALE(1,I)=SIGN(.00001,PSCALE(1,I))
      E=ALOG10(PSCALE(1,I))
      N=E
      IF(E.LT.0.) N=E-1.
      C=E-N
```

```
      D=10.**C
      IF((1..LT.D).AND.(D.LT.2.)) D=2.
      IF((2..LT.D).AND.(D.LT.2.5))D=2.5
      IF((2.5.LT.D).AND.(D.LT.5.))D=5.
      IF(5..LT.D) D=10.
      PSCALE(1,I)=D*10.**N
      IF(PSCALE(2,I).GT.0.) PSCALE(2,I)=0.
      IF(PSCALE(2,I).EQ.0.) GO TO 300
      PSCALE(2,I)=-PSCALE(1,I)
00300  CONTINUE
      DO 350 J=1,14
      SCALE(1,J)=PSCALE(1,J)
      SCALE(2,J)=PSCALE(2,J)
      IF(SCALE(1,J).EQ.SCALE(2,J))SCALE(1,J)=SCALE(2,J)+1.
00350 CONTINUE
00400 CONTINUE
       DO 10 I=1,6
       DATHD1(I)=DATHD(I)
      DATHD2(I)=DATHD(6+I)
       DATHD3(I)=DATHD(12+I)
       DATHD4(I)=DATHD(18+I)
       DATHD5(I)=DATHD(24+I)
       DATHD6(I)=DATHD(30+I)
00010  CONTINUE
      K=0
      DO 11 I=1,4
00011 DATHD7(I)=DATHD(36+I)
      CALL PLOTS(IBUF,1024,1)
      CALL FACTOR(FAC)
      CALL PLOT(0.,40.,-3)
      CALL PLOT(0.,-1.,-3)
      CALL SYMBOL(2.,0.,.15,DATHD1,0.,60)
      CALL SYMBOL(2.,-.25,.15,DATHD2,0.,          60)
      CALL SYMBOL(2.,-.5,.15,DATHD3,0.,60)
      CALL SYMBOL(2.,-.75,.15,DATHD4,0.,60)
      CALL SYMBOL(2.,-1.,.15,DATHD5,0.,60)
      CALL SYMBOL(2.,-1.25,.15,DATHD6,0.,60)
      CALL SYMBOL(2.,-1.5,.15,DATHD7,0.,70)
      DO 110 J=1,14
      IF(SCALE(1,J).EQ.SCALE(2,J)) SCALE(1,J)=SCALE(2,J)+1.
       IF(NPLOT(J).EQ.0) K=K+1
       IF(NPLOT(J).EQ.0) GO TO 110
      IF(NPLOT(J).EQ.1) DATTIT(J)=TPRNTA(IDAT(J))
      IF(NPLOT(J).EQ.2)DATTIT(J)=TPRNTB(IDAT(J))
       YTIT=-2.787-1.824*(J-K-1)
        IF(J.GE.9) YTIT=-3.287-1.824*(J-K-1)
      DATT=DATTIT(J)
      CALL SYMBOL(0.,YTIT,.15,DATT,0.,10)
       CALL NUMBER(1.,YTIT+.787,.15,SCALE(1,J),0.,2)
       CALL NUMBER(1.,YTIT-.787,.15,SCALE(2,J),0.,2)
       CALL PLOT(2.,YTIT-.787,3)
       CALL PLOT(2.+XMMAX,YTIT-.787,2)
       CALL PLOT(2.+XMMAX,YTIT+.787,2)
       CALL PLOT(2.,YTIT+.787,2)
       CALL PLOT(2.,YTIT-.787,2)
```

```
      CALL PLOT(2.,YTIT,2)
      CALL PLOT(2.+XMMAX,YTIT,2)
      CALL PLOT(2.,YTIT,3)
      JM=TMMAX+1.
      DO 20 I=1,JM
      CALL PLOT(2.+CHTSP*I,YTIT+.1,3)
      CALL PLOT(2.+CHTSP*I,YTIT-.1,2)
00020   CONTINUE
      XO=2.
      XIP=2.
     YO=YTIT-.787+(DATPLT(1,J)-SCALE(2,J))*1.574/(SCALE(1,J)
    1  -SCALE(2,J))
      CALL PLOT(XO,YO,3)
      NCNT1=NCNT-1
      DO 40 I=2,NCNT1
      XI=2.+CHTSP*DATPLT(I,15)
      IF(XI.LT.XIP) XI=XIP
      XIP=XI
      IF(DATPLT(I,J).GT.SCALE(1,J)) DATPLT(I,J)=SCALE(1,J)
      IF(DATPLT(I,J).LT.SCALE(2,J)) DATPLT(I,J)=SCALE(2,J)
      YI=YTIT-.787+(DATPLT(I,J)-SCALE(2,J))*1.574/(SCALE(1,J)
    1  -SCALE(2,J))
     CALL PLOT(XI,YI,2)
00040 CONTINUE
     CALL PLOT(0.,YTIT-1.824,3)
00110 CONTINUE
     XP=6.+XMMAX
     IF(XP.LT.12.)XP=12.
     CALL PLOT(XP,0.,999)
     RETURN
     END

**********************************************************************

     SUBROUTINE SHOW
     COMMON/XFLOAT/A(500)/IFIXED/IA(200)
      COMMON/CHF/B(200)/ICHF/IB(50)
     COMMON/DATA/D(300)
     DIMENSION ICHNBF(4,3)
     DIMENSION PAR(17)
     DIMENSION NUM(3)
     DIMENSION AV(17,3),STD(17,3),XMIN(17,3),OMAX(17,3)
      DIMENSION DPOINT(11)
      DIMENSION HOTOKE(7)
      DIMENSION ITOUCH(5)
      EQUIVALENCE(ITOUCH(1),IA(10))
      EQUIVALENCE(JLAND,IB(50))
     EQUIVALENCE (A(0001),PHI)
     EQUIVALENCE(A(2),THET)
     EQUIVALENCE(A(3),PSI)
     EQUIVALENCE(A(25),ALFA)
     EQUIVALENCE(A(26),BETA)
     EQUIVALENCE(A(35),GAMV)
```

```
      EQUIVALENCE(A(75),VEQ)
      EQUIVALENCE(A(80),ALTD)
      EQUIVALENCE(A(231),THETIC)
      EQUIVALENCE(A(232),PSIIC)
      EQUIVALENCE(A(238),VEQIC)
      EQUIVALENCE(A(359),R2D)
      EQUIVALENCE(QR,A(38))
       EQUIVALENCE(VHW,A(70)),(THTN,B(17))
       EQUIVALENCE(THROT,B(14)),(RLONSTK,B(7)),(HE,B(140)),(XE,B(141))
       EQUIVALENCE(YE,B(142)),(XCG,A(174)),(YCG,A(175))
       EQUIVALENCE(AIL,B(2)),(RUDPED,B(11)),(RLATSTK,B(9)),(ALTCOM,B(143
000001  )),(ALTDCOM,B(144)),(RLNSTKO,B(8)),(THROTIC,B(15)),(ALFAIC,A(42)
000002  ),(TOTRM,B(20)),(YCOM,B(145)),(XCOM,B(146)),(ITOUCH(1),IA(10))
      EQUIVALENCE(ALT,A(83)),(RLTSTKO,B(10)),(VX,A(200)),(VY,A(201))
       EQUIVALENCE(PSICOM,B(150)),(PHICOM,B(148)),(THETCOM,B(147))
       EQUIVALENCE(VXCOM,B(149))
       EQUIVALENCE(ALTCOM,B(143)),(ALTDCOM,B(144))
      EQUIVALENCE(IA(1),IMODE)
       EQUIVALENCE(IOVDK,IA(15))
       EQUIVALENCE(IMOV,IB(1)),(ICON,IB(2)),(ITRANS,IB(6))
       EQUIVALENCE(ILAND,IB(3))
      EQUIVALENCE(IA(83),ISHOW)
      EQUIVALENCE(IA(84),INUMBR)
      EQUIVALENCE(IA(90),ICHNBF(1,1))
       EQUIVALENCE(HOTDKE(1),A(447))
      EQUIVALENCE(IA(105),NRUN)
      EQUIVALENCE(B(20),TOTRM)
      EQUIVALENCE(IB(4),IZERO)
      EQUIVALENCE(IB(10),ISHPRT)
      EQUIVALENCE(IB(011),ICOPY)
      EQUIVALENCE(IB(014),ITD)
      EQUIVALENCE(IB(017),IUSER)
       EQUIVALENCE(D(250),NUM(1))
      EQUIVALENCE(D(04),PAR(1))
      EQUIVALENCE(D(021),AV(1))
      EQUIVALENCE(D(072),STD(1))
       EQUIVALENCE(D(123),XMIN(1))
      EQUIVALENCE(D(174),QMAX(1))
      EQUIVALENCE(D(225),DPOINT(1))
      EQUIVALENCE(D(240),NEXT)
      EQUIVALENCE(UTURB,A(61)),(VTURB,A(62)),(WTURB,A(63))
      EQUIVALENCE(XE1,A(499)),(YE1,A(498))
      EQUIVALENCE(ALTD,A(80)),(ALTDCOM,B(144))
      DATA (ICHNBF(1,L),L=1,3)/3,4,5/
      DATA (ICHNBF(2,L),L=1,3)/11,12,13/
      DATA (ICHNBF(3,L),L=1,3)/19,20,21/
       DATA ITOP/10/
      IF(IMODE)10,900,100
00010 CONTINUE
      IDA=0
      DO 11 I=1,3
      NUM(I)=0
      DO 11 J=1,17
      AV(J,I)=0.0
      STD(J,I)=0.0
```

```
      XMIN(J,I)=1.0E9
      QMAX(J,I)=-1.0E9
00011 CONTINUE
      DO 12 J=1,11
      DPOINT(J)=0.0
00012  CONTINUE
00100 CONTINUE
       ITD=0
       IF(JLAND)2,2,1
00001  ITD=0
       DO 3 J=1,4
      IF(ITOUCH(J).EQ.1)ITD=1
00003 CONTINUE
00002 CONTINUE
      IMOV=0
      ICON=0
      ITRANS=0
      IF(VRW.LT.100..AND.THTN.GT.60..AND.ITDP.EQ.0)IMOV=1
      IF(VRW.GT.100..AND.THTN.GT.20..AND.ITDP.EQ.0)ITRANS=1
      IF(VRW.GT.100..AND.THTN.LT.20..AND.ITDP.EQ.0)ICON=1
      MODEP=IMODE
      IF(ICON)102,102,101
00101 CONTINUE
      K=1
      GO TO 401
00102 CONTINUE
      IF(ITRANS)301,301,201
00201 CONTINUE
      K=2
      GO TO 401
00301 IF(IMOV)609,609,302
00302 K=3
00401 CONTINUE
      PAR(1)=HLONSTK-PLNSTKO
      PAR(2)=QH
      PAR(3)=THROT-THROTIC
      PAR(4)=RLATSTK-RLTSTKO
      PAR(5)=HUDPED
      PAR(6)=XEI
      PAR(7)=YEI
      PAR(8)=ALTD-ALTDCOM
      PAR(9)=UTURB
      PAR(10)=VTURB
      PAR(11)=VX-VXCOM
      PAR(12)=THET-THETCOM
      PAR(13)=HETA
      PAR(14)=PSI-PSICOM
      PAR(15)=PHI-PHICOM
      PAR(16)=ALT-ALTCOM
      PAR(17)=WTURB
      DO 402 J=1,17
      AV(J,K)=AV(J,K)+PAR(J)
      STD(J,K)=STD(J,K)+PAR(J)**2
      IF(PAR(J).GT.QMAX(J,K))QMAX(J,K)=PAR(J)
      IF(PAR(J).LT.XMIN(J,K))  XMIN(J,K)=PAR(J)
```

```
00402 CONTINUE
      NUM(1)=NUM(1)+ICON
      NUM(2)=NUM(2)+ITRANS
      NUM(3)=NUM(3)+IHOV
00501 CONTINUE
      IF(JLAND.EQ.0) GO TO 609
      IF(IOVOK) 601,601,502
00502 CONTINUE
      IF(IDA) 503,503,601
00503 CONTINUE
      DPOINT(1)=THET
      DPOINT(2)=ALT-ALTCOM
      DPOINT(3)=YE
      IDA=1
00601 CONTINUE
      IF(ITD)609,609,603
00603 CONTINUE
      DPOINT(4)=XE
      DPOINT(5)=YE
      DPOINT(6)=THET
      DPOINT(7)=ALFA
      DPOINT(8)=PSI
      DPOINT(9)=PHI
      DPOINT(10)=BETA
      DPOINT(11)=ALTD-HOTDKE(2)
      ITD=1
00609 CONTINUE
      ITDP=ITD
00900 CONTINUE
      RETURN
      END
```

```
****************************************************************************

       SUBROUTINE DATSAV
       COMMON/XFLOAT/A(500)/IFIXED/IA(200)
       COMMON/CHF/B(200)/ICBF/IB(50)
       COMMON/DAT/DATTIT(15),DATHD(40),CHTSP,SCALE(28),
     1  THMAX,NPLOT(14),ISCALE
       COMMON/DAT2/DATPLT(300,15)
      COMMON/DPLOCK/IPLOT(14),IPRNT(11)
      COMMON/CONEM/EMCONV(15),PSCALE(2,14)
      COMMON/SCP/MTHANS(14),MPLOT(14),IDAT(14)
       EQUIVALENCE(IA(1),IMODE)
      EQUIVALENCE(NCNT,IB(36))
       EQUIVALENCE(RTIME,A(304))
      EQUIVALENCE(A(303),TIME)
      DATA(IPLOT(I),I=1,14)/122007,122009,122011,12016,12014,
     1   11002,11001,11003,51070,51083,51080,52140,11025,52142/
       DATA ISCALE/0/
       DATA ISET/0/
       IF(IMODE)1,1,6
00001  IF(ISET)2,2,1000
```

```
00002  DO 3 J=1,15
        DO 3 I=1,300
00003   DATPLT(I,J)=0.
       DO 7 J=1,14
       PSCALE(1,J)=-1.E9
       PSCALE(2,J)=1.E9
00007  CONTINUE
       DO 100 J=1,14
       MTRANS(J)=IPLOT(J)/10000
       MPLOT(J)=(IPLOT(J)-MTRANS(J)*10000)/1000
       IDAT(J)=IPLOT(J)-MTRANS(J)*10000-MPLOT(J)*1000
00100  CONTINUE
        ISET=1
        NCNT=0
        TMMAX=0.
        TIM2=0.
        GO TO 1000
00006   ISET=0
        DTIM=RTIME/300.
        IF(RTIME.LE.15.) DTIM=.05
        IF(TIME-TIM2) 1000,4,4
00004    TIM2=TIM2+DTIM
        NCNT=NCNT+1
        IF(NCNT.GT.300) GO TO 1000
       DO 200 J=1,14
       IF(MPLOT(J).EQ.1) DATPLT(NCNT,J)=A(IDAT(J))*EMCONV(MTRANS(J))
       IF(MPLOT(J).EQ.2) DATPLT(NCNT,J)=B(IDAT(J))*EMCONV(MTRANS(J))
       IF(DATPLT(NCNT,J).GT.PSCALE(1,J))   PSCALE(1,J)=DATPLT(NCNT,J)
       IF(DATPLT(NCNT,J).LT.PSCALE(2,J))PSCALE(2,J)=DATPLT(NCNT,J)
00200  CONTINUE
        DATPLT(NCNT,15)=TIME
        TMMAX=TIME
01000   CONTINUE
        IF(NCNT.GE.300) NCNT=300
        RETURN
       END
```

****

```
       SUBROUTINE PRINTO
       COMMON/XFLOAT/A(500)/IFIXED/IA(200)
        COMMON/CHF/B(200)/ICBF/IB(50)
       COMMON/DATA/D(300)
       DIMENSION AUM(11,2)
       DIMENSION BUT(11,2)
       DIMENSION AV(17,3),STD(17,3),XMIN(17,3),QMAX(17,3)
        DIMENSION DATE1(2)
       DIMENSION DPOINT(11)
       DIMENSION HI(17,2)
       DIMENSION HJ(4,4)
       DIMENSION HK(11,2)
       DIMENSION KC(6)
       DIMENSION KD(6)
```

```
      DIMENSION NUM(3)
      DIMENSION OUT(8,3)
       DIMENSION PAR(17)
      DIMENSION SUM(8,3)
       EQUIVALENCE(IA(1),IMODE)
      EQUIVALENCE(IA(083),ISHOW)
      EQUIVALENCE(IA(084),INUMBR)
      EQUIVALENCE(IA(90),ICHNHF)
      EQUIVALENCE(IA(105),NRUN)
      EQUIVALENCE(IB(4),IZERO)
      EQUIVALENCE(IB(10),ISHPRT)
      EQUIVALENCE(IB(011),ICOPY)
      EQUIVALENCE(IB(014),ITD)
      EQUIVALENCE(IB(015),IDA)
        EQUIVALENCE(D(250),NUM(1))
      EQUIVALENCE(D(04),PAR(1))
      EQUIVALENCE(D(021),AV(1))
      EQUIVALENCE(D(072),STD(1))
       EQUIVALENCE(IA(199),N2)
      EQUIVALENCE(D(123),XMIN(1))
      EQUIVALENCE(D(174),QMAX(1))
      EQUIVALENCE(D(225),DPOINT(1))
      EQUIVALENCE(D(240),NEXT)
      EQUIVALENCE(TAF,A(493)),(TVF,A(492)),(TSCPSEC,A(491))
      EQUIVALENCE(PEDAF,A(490)),(PEDVF,A(489)),(PEDSC,A(488))
      EQUIVALENCE(RLNAF,A(487)),(RLNVF,A(486)),(RLNSC,A(485))
      EQUIVALENCE(RLTAF,A(484)),(RLTVF,A(483)),(RLTSC,A(482))
      DATA ICOPY/1/
       DATA (KC(I),I=1,6)/1,4,7,10,13,16/
      DATA (KD(I),I=1,6)/3,6,9,12,15,17/
       DATA(HI(1,I),I=1,2)/4HDLON ,4HSTK  /
       DATA(HI(02,I),I=1,2)/4HQB     ,4H   /
       DATA(HI(03,I),I=1,2)/4HDTHR   ,4HT   /
       DATA(HI(4,I),I=1,2)/4HDLAT ,4HSTK  /
       DATA(HI(05,I),I=1,2)/4HRUDP ,4HED  /
      DATA(HI(6,I),I=1,2)/4HXF1  ,4H     /
      DATA(HI(7,I),I=1,2)/4HYF1  ,4H     /
      DATA(HI(8,I),I=1,2)/4HALTD,4HE      /
      DATA(HI(9,I),I=1,2)/4HUTUR,4HB    /
      DATA(HI(10,I),I=1,2)/4HVTUR,4HB     /
       DATA(HI(11,I),I=1,2)/4HVXE ,4H      /
       DATA(HI(12,I),I=1,2)/4HTHET ,4HE     /
        DATA(HI(13,I),I=1,2)/4HBETA ,4H    /
       DATA(HI(14,I),I=1,2)/4HPSIE ,4H    /
       DATA(HI(15,I),I=1,2)/4HPHIE ,4H    /
       DATA(HI(16,I),I=1,2)/4HALTE ,4H    /
      DATA(HI(17,I),I=1,2)/4HWTUR    ,4HB     /
        DATA(HJ(1,I),I=1,4)/4HTHET ,4H MA  ,4HXIMU  ,4HM      /
      DATA (HJ(2,I),I=1,4)/4HTHET,4HA MI   ,4HNIMU  ,4HM     /
      DATA (HJ(3,I),I=1,4)/4HZF    ,4H  MA,4HXIMU,4HM     /
      DATA(HJ(4,I),I=1,4)/4HZF    ,4H  MI  ,4HNIMU  ,4HM     /
      DATA(HK(01,I),I=1,2)/4HTHET,4HA     /
       DATA(HK(2,I),I=1,2)/4H  HE,4H      /
       DATA(HK(3,I),I=1,2)/4H  YE  ,4H     /
        DATA(HK(4,I),I=1,2)/4H  XE  ,4H     /
```

```
      DATA(HK(5,I),I=1,2)/4H  YE  ,4H    /
      DATA(HK(06,I),I=1,2)/4HTHET,4HA   /
      DATA(HK(07,I),I=1,2)/4HALFA  ,4H    /
      DATA(HK(08,I),I=1,2)/4H PSI  ,4H    /
      DATA(HK(09,I),I=1,2)/4HPHI  ,4H    /
      DATA(HK(10,I),I=1,2)/4HBETA,4H     /
      DATA(HK(11,I),I=1,2)/4HALTD ,4HE    /
      DATA NHFC/2/
      IF(IZERO) 201,201,101
00101 CONTINUE
      ANO=0.0
      DO 102 I=1,3
      DO 102 J=1,8
      SUM(J,I)=0.0
      OUT(J,I)=0.0
102   CONTINUE
      DO 103 I=1,2
      DO 103 J=1,11
      AUM(J,I)=0.0
      BUT(J,I)=0.0
103   CONTINUE
      RETURN
201   CONTINUE
301   CONTINUE
      DATE1(1)=DATE(D)
      DATE1(2)=TIME(D)
      IF(ITD.NE.1.OR.IDA.NE.1.) GO TO 304
      ANO=ANO+1.
      DO 302 I=1,3
      SUM(1,I)=SUM(1,I)+QMAX(10,I)
      SUM(2,I)=SUM(2,I)+XMIN(10,I)
      SUM(3,I)=SUM(3,I)+QMAX(16,I)
      SUM(4,I)=SUM(4,I)+XMIN(16,I)
      SUM(5,I)=SUM(5,I)+QMAX(10,I)**2
      SUM(6,I)=SUM(6,I)+XMIN(10,I)**2
      SUM(7,I)=SUM(7,I)+QMAX(16,I)**2
      SUM(8,I)=SUM(8,I)+XMIN(16,I)**2
      OUT(1,I)=SUM(1,I)/ANO
      OUT(2,I)=OUT(2,I)/ANO
      OUT(2,I)=SUM(2,I)/ANO
      OUT(3,I)=SUM(3,I)/ANO
      OUT(4,I)=SUM(4,I)/ANO
      OUT(5,I)=SQRT(SUM(5,I)/ANO-OUT(1,I)**2)
      OUT(6,I)=SQRT( SUM(6,I)/ANO-OUT(2,I)**2)
      OUT(7,I)=SQRT(SUM(7,I)/ANO-OUT(3,I)**2)
      OUT(8,I)=SQRT(SUM(8,I)/ANO-OUT(4,I)**2)
302   CONTINUE
      DO 303 J=1,11
      AUM(J,1)=AUM(J,1)+DPOINT(J)
      AUM(J,2)=AUM(J,2)+DPOINT(J)**2
      BUT(J,1)=AUM(J,1)/ANO
      BUT(J,2)=SQRT(AUM(J,2)/ANO-BUT(J,1)**2)
303   CONTINUE
304   CONTINUE
      DO 4002 J=1,17
```

```
      DO 4002 I=1,3
       IF(NUM(I).EQ.0.) GO TO 305
      AV(J,I)=AV(J,I)/NUM(I)
       STD1=STD(J,I)/NUM(I)-AV(J,I)**2
       IF(STD1.LT.0.) GO TO 306
      STD(J,I)=SQRT(STD(J,I)/NUM(I)-AV(J,I)**2)
       GO TO 307
 305  AV(J,I)=0.
 306  STD(J,I)=0.
  307 CONTINUE
 4002  CONTINUE
      KRUN=N2
      DO 6000 LL=1,ICOPY
      IPAGE=1
      PRINT 5000
       PRINT 5001,KRUN,DATE1(1),DATE1(2),IPAGE
      PRINT 5021
      PRINT 5025
      PRINT 5022,NUM(1),NUM(2),NUM(3)
      PRINT 5023
      DO 6001 MO=1,6
      IC=KC(MO)
      ID=KD(MO)
      PRINT 5016
      PRINT 5024,(((HI(I,J),J=1,2),(AV(I,J),STD(I,J),QMAX(I,J)
000001 ,XMIN(I,J),J=1,3),I=IC,ID))
 6001 CONTINUE
      IPAGE=IPAGE+1
      PRINT 5011
       PRINT 5001,KRUN,DATE1(1),DATE1(2),IPAGE
      PRINT 5027,ANO
      PRINT 5033
      PRINT 5020
      PRINT 5017
      PRINT 5029,(HJ(1,J),J=1,4),OUT(1,1),OUT(5,1),OUT(1,2),OUT(5,2)
000001 ,OUT(1,3),OUT(5,3)
      PRINT 5029,(HJ(2,J),J=1,4),OUT(2,1),OUT(6,1),OUT(2,2),OUT(6,2)
000001  ,OUT(2,3),OUT(6,3)
      PRINT 5029,(HJ(3,J),J=1,4),OUT(3,1),OUT(7,1),OUT(3,2),OUT(7,2)
000001  ,OUT(3,3),OUT(7,3)
      PRINT 5029,(HJ(4,J),J=1,4),OUT(4,1),OUT(8,1),OUT(4,2),OUT(8,2)
000001  ,OUT(4,3),OUT(8,3)
      PRINT 5016
      PRINT 5030
      PRINT 5016
      PRINT 5031,((HK(I,J),J=1,2),DPOINT(I),BUT(I,1),BUT(I,2),I=1,3)
      PRINT 5016
      PRINT 5032
      PRINT 5016
      PRINT 5031,((HK(I,J),J=1,2),DPOINT(I),BUT(I,1),BUT(I,2),I=4,11)
      PRINT 5011
 6000 CONTINUE
      PRINT 7000
      PRINT 7001,TAF,TVF,TSCPSEC
      PRINT 7002,PEDAF,PEDVF,PEDSC
```

```
      PRINT 7003,RLNAF,RLNVF,RLNSC
      PRINT 7004,RLTAF,RLTVF,RLTSC
 7000 FORMAT(10X,30H PILOT WORK LOAD FUNCTIONS          )
 7001 FORMAT(10X,10H TAF=      ,F10.4,10H TVF=      ,F10.4,10HTSCPSEC=
     1     ,F10.4)
 7002 FORMAT(10X,10HPFDAF=     ,F10.4,10H PEDVF=    ,F10.4,10HPEDSC=
     1     ,F10.4)
 7003 FORMAT(10X,10HRLNAF=      ,F10.4,10HRLNVF=     ,F10.4,10HRLNSC=
     1     ,F10.4)
 7004 FORMAT(10X,10H RLTAF=   ,F10.4,10H RLTVF=   ,F10.4,10H RLTSC=
     1    ,F10.4)
09000 RETURN
05000 FORMAT(42X,17HCRF  FLIGHT  DATA    )
05001  FORMAT(12X,12HRUN NUMBER   ,I4,20X,6HDATE   ,2A10,16X,6HPAGE
000001  ,I4)
05011 FORMAT(1H1)
05016 FORMAT(1H0)
05017 FORMAT(1H )
05021 FORMAT(1H0,48X,17HSTATISTICAL  DATA     )
05022 FORMAT(1H0,15X,28HFIRST INTERVAL SAMPLE SIZE   ,I5
000001,  7X,28HSECOND INTERVAL SAMPLE SIZE   ,I5
000002 ,7X,28HTHIRD INTERVAL SAMPLE SIZE  ,I5)
05023  FORMAT(1H0,15X,4HMEAN,7X,3HRMS,3X,7HMAXIMUM,3X,7HMINIMUM
000001  ,  6X,4HMEAN,7X,3HRMS,3X,7HMAXIMUM,3X,7HMINIMUM
000002  ,  6X,4HMEAN,7X,3HRMS,3X,7HMAXIMUM,3X,7HMINIMUM)
05024  FORMAT(1H ,2A4, 1X,12F10.3)
05025  FORMAT(1H0,15X,10HCNVNTNAL        ,30X
000001  21HTRANSITION MODE          ,19X,
000002  24H HOVER MODE                     )
05026  FORMAT(1H0,60HRROR - PRINTOUT CALLED WHEN ISHPRT = 0       )
05027 FORMAT( 1H0,19X,48HCONSECUTIVE RUN AVERAGES AND STANDARD DEVIATION
0000015,20X,F3.0,2X,4HRUNS)
05028  FORMAT(1H0,29X,10H  AVERAGE    ,6X,10H    RMS
000001  ,  6X,10H   AVERAGE  ,6X,10H    RMS
000002,  6X,10H   AVERAGE,6X,10H    RMS   )
05029 FORMAT(10X,4A4,  4X,6(F10.4,6X))
05030 FORMAT(1H0,19X,18HVALUES AT THE RAMP  ,27X,7HAVERAGE ,13X,3HRMS )
05031  FORMAT(10X,2A4,6X,F10.4,30X,F10.4,6X,F10.4)
05032  FORMAT(1H0,19X,20HTOUCHDOWN PARAMETERS   )
05033  FORMAT(1H0,29X ,10HCNVNTNAL
000001,25X,21HTRANSITION MODE
000002   ,11X,24HHOVER MODE                    )
      END
```

```
**********************************************************************
```

```
      BLOCK DATA
      COMMON/CHF/A(200)/ICHF/IB(50)
      COMMON/TBLOCK/TPRNTA(500),TPRNTB(200)
      COMMON/FLOAT/A(500)/IFIXED/IA(200)
      DIMENSION TPA1(50),TPA2(50),TPA3(50),TPA4(50),TPA5(50),TPA6(50),
     1TPA7(50),TPA8(50),TPA9(50),TPA10(50),TPB1(50),TPB2(50),TPB3(50),
     2TPB4(50)
```

```
      DIMENSION DXPA(15),DYPA(15),DZPA(15)
      EQUIVALENCE(TPA1(1),TPRNTA(1)),(TPA2(1),TPRNTA(51)),
     1(TPA3(1),TPRNTA(101)),(TPA4(1),TPRNTA(151)),(TPA5(1),TPRNTA(201))
     2,(TPA6(1),TPRNTA(251)),(TPA7(1),TPRNTA(301)),(TPA8(1),TPRNTA(351))
     3,(TPA9(1),TPRNTA(401)),(TPA10(1),TPRNTA(451))
      EQUIVALENCE(TPRNTB(1),TPB1(1)),(TPRNTB(51),TPB2(1)),(TPRNTB(101),
     1TPB3(1)),(TPRNTB(151),TPB4(1))
      EQUIVALENCE(SPAN,A(181)),(CHORD,A(182)),(DT,A(168)),(AREA,A(180))
     1,(D2R,A(358)),(R2D,A(359))
      EQUIVALENCE(PTURB,A(52)),(QTURB,A(53)),(RTURB,A(54))
      EQUIVALENCE(A(97),G),(XMASS,A(130)),(DXPA(1),A(184))
      EQUIVALENCE(IB(11),ICOPY)
      EQUIVALENCE(DYPA(1),A(210)        ),(VDIC,A(229)),(PHIIC,A(230))
      EQUIVALENCE(THETIC,A(231)),(PSIIC,A(232)),(GAMVIC,A(233)),(GAMHIC
     1,A(234)),(PRIC,A(235)),(QRIC,A(236)),(RRIC,A(237)),(VEQIC,
     2A(238)),(XIC,A(239)),(YIC,A(240)),(HIC,A(241)),(XIXXIC,A(243))
      EQUIVALENCE(XIYYIC,A(244)),(XIZZIC,A(245)),(XIXZIC,A(246)),
     1(WFUELIC,A(247)),(WWATIC,A(248)),(  WAITO,A(251)),(DZPA(1),A(257))
      EQUIVALENCE(D2R,A(358)),(R2D,A(359)),(       THROTIC,B(15))
      EQUIVALENCE(THTNCIC,B(18)),(VS,B(30)),(VNWHIC,B(35))
      EQUIVALENCE(VEWHIC,B(36)),(BETAIC,B(37)),(ALFAIC,B(42))
      EQUIVALENCE(IFLAT,IA(6)),(IDT1,IA(61)),(IDT2,IA(62)),(IDT3,
     1IA(63)),(ICB,IA(64)),(IGEAR,IA(104)),(NRUN,IA(105)),(INDEXT,
     2IA(110)),(IMACH,IA(114)),(NUSED,IA(180)),(IEULR,IA(184)),
     3  (ITASK,IB(30))
      EQUIVALENCE(TFPJ1,B(49)),(TRPJ1,B(50)),(TRJ1,B(51))
      EQUIVALENCE(COURSE,A(110))
      EQUIVALENCE(TYAWJ1,B(52))
      EQUIVALENCE(B(53),WWDOT1)
      EQUIVALENCE(IFLAT,IA(6))
      DATA IFLAT/1/
      DATA WWDOT1/5.6/
      DATA ICOPY/1/
      DATA SPAN/25.25/,CHORD/7.95/,DT/.05/,AREA/201./,D2R/.01745/,
     1 R2D/57.3/
      DATA PTURB/0./,QTURB/0./,RTURB/0./
      DATA ALFAIC/8./
      DATA VNTURB/0./,VETURB/0./,VDTURB/0./,VNW/0./,VEW/0./,VDW/0./
      DATA IDT1/25/,IDT2/50/,IDT3/100/
      DATA COURSE/0./
      DATA VS/50./
      DATA VEWHIC/0./,VNWHIC/-10./
      DATA GAMVIC/0./,GAMHIC/0./
      DATA PRIC/0./,QRIC/0./,RRIC/0./
      DATA PHIIC/0./,THETIC/0./,PSIIC/0./
      DATA (DXPA(I),I=1,15)/      6.69,-4.5,-5.75,-5.75,8.,50.,18.88,
     125.95,31.83,11.46,50.46,49.9,34.925,34.925,28.51/
      DATA (DYPA(I),I=1,15)/ 0.,0.,8.36,-8.36,0.,0.,0.,0.,0.,0.,0.,0.,
     1  11.06,-11.08,0. /
      DATA (DZPA(I),I=1,15)/5.67,4.7,4.5,4.5,13.,5.,8.,7.75,7.429,
     1   0.,0.,10.04,0.,0.,8.015/
      DATA WFUELIC/500./,WWATIC/100./,TFPJ1/950./,TRPJ1/1110./
      DATA TRJ1/612./,TYAWJ1/650./
      DATA XIC/100./,YIC/100./
      DATA XIXXIC/4295./,XIYYIC/25356./,XIZZIC/27506./,XIXZIC/876./
```

```
      DATA ICG/1/
      DATA WAITO/12500./
      DATA ITASK/2/
      DATA INDEXT/4/
      DATA NUSED/2/
      DATA G/32.2/
      DATA IGEAR/1/
      DATA VDIC/0./
      DATA(TPA1(I),I=1,50)/10H PHI-DEG. ,10H THETA-DEG,10H PSI-DEG. ,
     1  10H PHIR-RAD   ,10H THETR-RAD    ,
     210H PSIR-RAD.,10HPHID-R/S  ,10H THED R/S ,10H PSID R/S ,
     310H SINE PHI ,10H COSINEPHI,10H SIN THET ,10H COS THET ,10H SINPSI
     4    ,10H COSINEPSI,10H T11       ,10H T21      ,10H T31      ,
     110H T12      ,10H T22      ,10H  T32     ,10H  T13     ,10H  T23
     1    ,10H  T33      ,10H ALFA-DEG.,10H BETA-DEG ,10H ALFAR-RAD,
     110H BETAR-RAD,10H ALFD-R/S.,10H BETD-R/S.,10H SALPH    ,10H CALPH
     1   ,10H SIN BETA ,10H COS BETA ,10H GAMV-RAD.,10H GAMH-RAD.  ,
     110H PH-RAD/S.,10H UA-R./S. ,10H RH-R./S. ,10H PL-R/S.  ,10H QL-R/S
     1   ,10H RL-RAD/S.,10H PLH-R/S. ,10H QLB-R/S  ,10H RLB-R/S. ,
     110H PT-R/S.  ,10H QT-R/S   ,10H RT-P/S.  ,10H PRWN-R/S.,10H QRWN-R
     1/S.  /
      DATA(TPA2(I),I=1,50)/10HRRWN-R/S. ,10H PTURB-R/S  ,
     110H QTURB-R/S,10H RTURB-R/S,10H PBD-R/S2 ,10H QBD-R/S2 ,10H RBD-R/
     1S2 ,10H UB-M./S. ,10H VB-M./S. ,10H WB-M./S. ,10H UTURB-M/S,10H VT
     1URB-M/S,10H WTURB-M/S,10H VN-M/S    ,10H VE-M/S    ,10H VD-M./S.  ,
     110H VEE-M/S  ,10H VT-M/S    ,10H VG-M/S    ,10H VRW-M/S  ,10H MACH-
     1NO. ,10H VNP-M/S   ,10H VER-M/S   ,10H VDR-M/S   ,10H VEQ-KNOTS,
     110H VNW-M/S   ,10H VEW-M/S   ,10H VDW-M/S  ,10H              ,10H ALTD-
     1M/S,10H XLOND-R/S,10H XLATD-R/S, 10H ALT-MTR   ,10H XLON-RAD ,
     110H XLAT-RAD ,10H SINLAT    ,10H COS LAT   ,10H VND-M/S2 ,10H VED-M/
     1S2 ,10H VDD-M/S2 ,10H AX-M/S2   ,10H AY-M/S2  ,10H AZ-M/S2  ,10H AX
     1P-M/S2 ,10H AYP-M/S2 ,10H AZP-M/S2 ,10H G-M/S2     ,10H          ,
     1  10H            ,10H           /
      DATA(TPA3(I),I=1,50)/10H VCAL-KTS ,10H               ,10H XPR-METER ,
     110H YPR-METER,10H HPR-METER,10H DNR-METER,10H DER-METER,10H RR-MET
     1ER ,10H RTV-METER,10H COURSE-R.,10HXLATR-RAD.,10HXLONR-RAD.,10HCLA
     1TR-RAD. ,10HSIN THETR ,10HCOS THETR ,10HXIXX-KG-M2,10HXIYY-KG-M2,
     110HXIZZ-KG-M2,10HXIXZ-KG-M2,10H XMC(1)    ,10H XMC(2)    ,10H XMC(3)
     1    ,10H XMC(4)    ,10H XMC(5)    ,10H XMC(6)    ,10H XMC(7)    ,
     110H XMC(8)    ,10H XMC(9)    ,10H XMC(10)    ,10H XMASS-KG ,10H CL
     1      ,10H  CD          ,10H  CX          ,10H  CY          ,10H  CZ       ,10H F
     1AX-NTN    ,10H FAY-NTN   ,10H FAZ-NTN   ,10H FEX-NTN   ,10H FEY-NTN   ,
     110H FEZ-NTN   ,10H FGZ-NTN   ,10H FGY-NTN   ,10H FGZ-NTN   ,10H FTZ-NT
     1N   ,10H FTY-NTN   ,10H FTZ-NTN   ,10HFNORTH-NTN,10HFEAST-NTN   ,
     110H FDOWN-NTN  /
      DATA(TPA4(I),I=1,50)/10H FG-NTN     ,10H CROLL MOM,10H CPITCHMOM ,
     110H CYAW MOM ,10H TAL-NT-M ,10HTAM-NT-M. ,10HTAN-NT-M   ,10HTEL-NT-
     1M
     1 ,10HTEM-NT-M   ,10HTEN-NT-M. ,10H TGL-NT-M ,10H TGM-NT-M ,10H TGN-
     1NT-    ,10H TTL-NT-M ,10H TTM-NT-M ,10H TTN-NT-M ,10H DT1-SEC  ,
     110H DT2-SEC   ,10H DT3-SEC   ,10H HR-METER ,10H              ,10H
     1    ,10H                ,10H XCG METER,10H YCG METER,10H HCG METER  ,
     110H WAIT-KG   , 10HQBAR-NT/M2,10HWBARC-N/M2,10H AREA-M2  ,10H SPAN-
     1M.  ,10H CHORD-M. ,10HRHO-KG/M3 ,10H XNOSG-M   ,10H XMAING-M ,10H X
     1RWNG-M   ,10H XLWNG-M   ,10H DXPLT-M   ,10H DXTAIL-M ,10H DXRMDRG-M,
```

```
1 10HXFRNTNOZ-M,10HXREARNOZ-M,10HDXPJET-M  ,.10HDXREARPJ-M,10HDXYAW
1JET-M,10HDXRWGJ-M  ,10HDXLWGJ-M  ,10H DXCG-M   ,10H             ,
110H VX-M/S      /
 DATA(TPA5(I),I=1,50)/10H VY-M/S   ,7*(10H          ),10H SOUND-M/S
1,10H DYNOSG-M ,10H DYMAIN-M ,10H DYRWNGG-M, 10HDYLWNG-G-M,10H DYPL
1T-M  ,10HDYTAIL-M  ,10HDYRMDRG-M  ,10HDYFRNTNZ-M ,10HDYREARNZ-M ,
110HDYFPTCHJ-M,10HDYRPTCHJ-M,10HDYYAWJ-M- ,10HDYHWG-J-M ,10H DYLWG
1J-M ,10HDYCG-M      ,10H XPILOT-M ,10H YPILOT-M ,10H ZPILOT-M ,10H
1        ,10H VDIC-M/S ,10H PHIIC-DEG,10H THETIC-D.,10H PSIIC-D.  ,
110H GAMVIC-D.,10H GAMHIC-D.,10HPBIC-R/S. ,10H QBIC-R/S.,10H RAIC-R
1/S.,10H VEQIC-KTS,10H XIC-METER,10H YIC-METER,10H HIC-METER,10HWAI
1TIC-KG ,10HXIXXIC KM2 ,10HXIYYIC KM2,10HXIZZIC KMS,10HXIXZIC KMS,
1                                                              10HWFU
1ELIC-KG,10HWWAITIC-KG,10H WFUEL-KG ,10H WWAT-KG  /
 DATA(TPA6(I),I=1,50)/10H WAITO-KG ,10H WSTORE-KG,10H            ,
110H DXCGO-M  ,10H DZCGO-M  ,10H CG-PCT    ,10H DZNOSG-M ,10HDZMNG-M
1   ,10HDZPWG-M    ,10HDZLWG-M    ,10HDZPLT-M    ,10HDZTAIL-M      ,
110HDZRMDRG-M ,10HDZFNOZ-M   ,10HDZRNOZ-M   ,10HDZFPTCHJ-M,10HDZRPTCH
1J-M,10HDZYAW-J-M ,10HDZPWG-J-M ,10HDZLWG-J-M ,10HZCG-CGR-M ,10HHDE
1CK(1)-M,10HHDECK(2)-M,10HHDECK(3)-M,10HHDECK(4)-M,10HHDECK(5)-M,2*
1(10H          ),10HXIGNG-M    ,10HXIGMG-M.  ,10HXIGRWG-M    ,
1                                                          10HXIGLWG-
1M  ,2*(10H          ),10HYIGNG-M    ,10HYIGMG-M    ,10HYIGRWG-M    ,
110HYIGLWG-M  ,2*(10H          ),10HHIGNG-M    ,10HHIGMG-M        ,
1  10HHIGRWG-M  ,10HHIGLWG-M  ,6*(10H          )  /
 DATA(TPA7(I),I=1,50)/2*(10H          ),10H TIME-SEC ,10H RTIME-SEC
1,27*(10H          ),10HTEMPRAT    ,10HPRESRAT      ,
110H          ,10H DTIM1-SEC,10H T1-SEC    ,14*(10H          )/
 DATA(TPA8(I),I=1,50)/7*(10H          ),10H D2R-RAD/D,10HR2D-D/RAD
1     ,4*(10H          )
1,10HRHOZ-KG/M3,10HHRHOZ-M      ,10HTAMR-DEG-K ,10HPAMB-N/M2       ,
110HTTOT-NTN   ,10H PTOT-N/M2,10H DELAT-K   ,10HSUTEMPR            ,
13*(10H          ),10HXMCC1-1/S.,10HXMCC2-1/S.,10HXMCC3-1/S ,10HXMC
1C4-1/S.,10HXMCC5-1/S.,10HXMCC6-1/S.,10HXMCC7-1/S.,10HEXMX-NMS.  ,
110HEXMY-NMS   ,10HEXMZ-NMS   ,6*(10H          ),10H STATE(1)     ,
110H STATE(2)  ,10H STATE(3)  ,10H STATE(4)  ,10H STATE(5)  ,10H STATE(
16)  ,10H CONT(1)   ,10H CONT(2)   ,10H CONT(3)   ,10H CONT(4)   /
 DATA(TPA9(I),I=1,50)/10H CONT(5)   ,10H CONT(6)   ,3*(10H          )
1,10H UHIC-M/S ,10H VHIC-M/S ,10H WHIC-M/S ,4*(10H          ),
110H UHD-M/S2 ,10H VHD-M/S2 ,10H WHD-M/S2 ,10HVTWN-M/S   ,10HVTWE-M/
1S   ,10H VTWD-M/S ,10HVNTURB-M/S,10HVETURB-M/S,10HVDTURB-M/S,
111*(10H          ),10HVDKNNOS-MS,10HVDNMN-M/S ,10HVDNRW-M/S   ,
110HVDNLWG-M/S,3*(10H          ),10HVDENS-M/S ,10HVDEMN-M/S   ,
1 10HVDERWG-M   ,10HVDELWG-M   ,3*(10H          ),10HHDTDCG-M
1   ,10HHDTNG-M/S   ,10HHDTDMG-M/S   ,10HHDTDRW-M/S   /
 DATA(TPA10(I),I=1,50)/10HHDTDLW-M/S,2*(10H          ),10H VR-M/S
1       ,46*(10H          )  /
 DATA(TPH1(I),I=1,50)/10H STAB-DEG.,10H AIL-DEG  ,10H RUD-DEG.  ,
110H          ,10H STAB1-DEG,10H PSAS-D    ,10HRLONSTK-CM,10HRLNSTKO
1-CM ,10HRLATSTK-CM,10HRLTSTKO-CM,10HRUDPED-CM ,10H YSAS-DEG  ,10HR
1SAS-DEG   ,10HTHROT-DEG ,10HTHROTIC-D.,10HTHTNC-DEG.,10HTHTN-DEG   ,
110HTHTNIC-DEG,10HRN1-PCT    ,10H TOUT-NTN ,10HPEDO-CM   ,10HTHRTMN-
1DEG,10HTHRTMX-DEG,10HTHTMN-DEG ,10HTHTMX-DEG ,10HPHIMN-DEG ,10HPHI
1MX-DEG ,10H PEDMN-CM ,10H PEDMX-CM ,10H VS-M/S    ,2*(10H
1),10HTHTNMN-DEG,10HTHTNMX-DEG,10HVNWBIC-M/S,10HVEWHIC-M/S,10HRETAI
```

```
1(-DEG,10HSTKMIN-CM ,10H STKMAX-CM,10HRLSMIN-CM ,10HRLSMAX-CM ,
110H ALFAIC-D.,10H          ,10H TFPJ-NTN ,10HTRPJ-NTN  , 10HTYAWJ
1-NTN   ,10H TRRJ-NTN ,10H TLRJ-NTN ,10H TFPJ1-NTN,10HTRPJ1-NTN /
 DATA(TPR2(I),I=1,50)/10H TRJ1-NTN ,10HTYAWJ1-NTN ,10HWWDOT2-KG   ,
12*(10H          ),10H TAUENG-S.,10HRDRAG-NTN ,10HTFN-NTN   ,
110H TRN-NTN   ,10HVJRCS-M/S ,10H UGE-PCT  ,10H          ,10HGEF1
1    ,10H GEF2-1/M.,10H ALTGE-NTN,10HFCR-KG/S. ,10HALRJ-CM2  ,
110HAFPJ-CM2  ,10HARPJ-CM2  ,10H ARRJ-CM2 ,10H AYAWJ-CM2,10H CSPLAY
1F   ,10HCSPLAYR    ,10H CRJ        ,10H CFPJA      ,10H CRPJA      ,10HSFP
1JA      ,10H SRPJA      ,10H SRJ         ,10H RLAM1-RAD,10H RLAM2        ,
110H PHIS-RAD.,10H THETS-RAD,10H PSIS-RAD ,10H THETDSR/S,
110HPHISD-R/S ,10HPSISD-R/S ,10HHEAVE-M.   ,10HHEAVED-M/S,10HSURGE-M
1.      ,10HSURGED-M/S,10H SWAY-M     ,10HSWAYD-M/S ,10H X1S-M         ,
110H X2S-METER,10H Y1S-METER,10H Y2S-METER,10H X3S-METER,10H H3S-ME
1TER ,10H H1S-METER/
 DATA(TPR3(I),I=1,50)/10H XTD-METER,10H YTD-METER,10HH1PS-METER,
110HHMAG-METER,10HPHIROL-RAD,10HPHIHV-RAD ,10HOMEGH-R/S ,10HOMEGH-R
1/S ,10HROLLM-RAD ,10HPITCHM-RAD,10HOMEGP-R/S ,10H PHIP-RAD ,10HYAW
1M-RAD   ,10HOMEGY-R/S ,10HPHIY-RAD   ,10H SURGM-M.   ,10HOMEGSRG-RS.
110HPHISRG-R. ,10HSWAYM-M.    ,10HOMEGSWY-RS.10HPHISWY-R. ,10H VJR
1    ,10H CDQ 1/RAD,10HCLADOT-1/R,10H CLP-1/RAD,10HCMQ-1/RAD ,10HCYD
1R-1/D. ,10H CYR-1/RAD,10HCLDA-1/D. ,10HCLQ-1/RAD ,10HCMADT-1/R ,
110HCNDR-1/D. ,10HCNR-1/RAD ,10HCYDA-1/DEG, 10HCYP-1/RAD ,4*(10H
1           ),10H HE-METER ,10H XE-METER ,10H YE-METER  ,10HALTCOM-M  ,
110HALTDCOM-MS,10HYCOM-M.     ,10HXCOM-M.      ,10HTHETCOM-D.,10HPHICOM-
1DEG,10HVXCOM-M/S ,10HPSICOM-D.  /
 DATA(TPR4(I),I=1,50)/2*(10H                    ),10HUGST-M2/S2,10HVGST-M2/S2
1,10HWGST-M2/S2,10HXFRZ-M.    ,10HSWBSW        ,10H SUHSW      ,10H
1    ,10H RKQ-D/R/S,10HTLEADP-S   ,10HPSAS1-DEG ,10HPSAS1I-D-S      ,
110H               ,10HRBWSHI-D-S,10HRBWSH-D.   ,10HRKDRRB-DRS,10HTLAGRB-
1S   ,10HRPDRA DMS2,10HTLEADAY-S.,10HTLAGAY-S.   ,10HAYFILTI-D.     ,
110HAYFILT-D. ,10HYSAS2-D.   ,10HRINCT-D.    ,10HRKDRDA-D/D,10HYSAS1-D
1    ,10HYSASLIM-D ,10HRPHDA-D/RS,10HTLEADDA-S.,10HTLAGDA-S. ,10HPBW
1SH-D.   ,10HPWSHI-D.   ,10HPSASLIM-D.,10H AILC-D.    ,10HTAUACDA-S. ,
110HTAUCHS-S. ,10H RUD2-D.    ,10H XTD1-M.    ,10H YTD1-M.    ,10HTAUALPH
1-S.   ,4*(10H           )  /
 END
```

••

```
C       TITLE                                   RICPRINT
        SUBROUTINE RICPRI(TITLE,A1,A2,A3,A4,A5,A6,A7,A8,A9,NODAT,U)
C
C
        COMMON /XFLOAT/A(500)/IFIXED/IA(200)
C
C   AUTHORS  -M.STEWART AND D. ASTILL-  COMPUTER SCIENCES CORPORATION
C
C
C   INFORMATION SUPPLIED IN CALLING SEQUENCE
C
C
C       TITLE ... IS WORD CONTAINING 4 HOLLERITH CHARACTERS FOR PRINTOUT
C       A1  *                                                    HEADING
C       A2  *
```

```
C     .      **  VARIABLES TO BE PRINTED
C     .      *
C     A9   *
C     NODAT ... ARRAY(DIMENSIONED 18) TWO WORDS TO EACH VARIABLE.  EACH
C             PAIR CONTAINS 8 HOLLERITH CHARACTERS TO BE USED AS
C             VARIABLE NAME ON PRINTOUT
C     U     ... ARRAY (DIMENSIONED 9) EACH WORD CONTAINS 4 HOLLERITH
C      CHARACTERS DESCRIBING UNITS OF VARIABLE
C
C
C
      EQUIVALENCE (A(1),PHI),(A(2),THET),(A(3),PSI),(A(25),ALFA)
     1,(A(26),BETA),(A(29),ALFD),(A(30),BETD),(A(35),GAMV)
     2,(A(36),GAMH),(A(37),PB),(A(38),QB),(A(39),RB),(A(55),PBD)
     3,(A(56),QBD),(A(57),RBD),(A(64),VN),(A(65),VE),(A(66),VD)
     4,(A(70),VPW),(A(71),XMACH),(A(75),VEO),(A(88),VND),(A(89),VED)
     5,(A(90),VDD),(A(116),XIXX),(A(117),XIYY),(A(118),XIZZ)
     6,(A(119),XIXZ),(A(131),CL),(A(132),CD),(A(133),CX),(A(134),CY)
     7,(A(135),CZ),(A(136),FAX),(A(137),FAY),(A(138),FAZ),(A(152),CLL)
     8,(A(153),CLM),(A(154),CLN),(A(155),TAL),(A(156),TAM)
     9,(A(157),TAN),(A(177),WAIT),(A(180),AREA),(A(181),SPAN)
     A,(A(182),CHORD),(A(256),CG)
       EQUIVALENCE (FFX,A(139)),(FEY,A(140)),(FEZ,A(141)),(FGX,A(142))
     1,(FGY,A(143)),(FGZ,A(144)),(FTX,A(145)),(FTY,A(146))
     2,(FTZ,A(147)),(TEL,A(158)),(TEM,A(159)),(TEN,A(160))
     3,(TGL,A(161)),(TGM,A(162)),(TGN,A(163)),(TTL,A(164))
     4,   (TTM,A(165)),(TTN,A(166))
       EQUIVALENCE (XCG,A(174)),(YCG,A(175)),(HCG,A(176))
       EQUIVALENCE (R2D   ,A(359))
       EQUIVALENCE(NRUN,IA(105))
       EQUIVALENCE(IA(199),N2)
C
C
       DIMENSION NQDAT(18),U(9)
C
       DIMENSION UDAT(15),ODAT(85)
C
        DATA UDAT/4HDEG.,4HR/S ,4HR/S2,4HNTN.,4HN-M.,4HM/S.,4HM/S2,
     14HMETR,4HKGM2,4HMTR2,4HPCT ,4HKTS ,4HKG   ,4H    ,4H    /
C
       DATA UDAT
     1/4H  PH, 4HI   ,4H  TH,4HET  ,4H  PS,4H  AL,4HFA  ,4H  BE,4HTA  ,
     24H  GA,4HMV  ,4HFD  ,4HTD  ,4HMH  ,4H  PB,4H    ,4H  QB,4H  RB,
     34HD   ,4H  CD,4H  CL,4H  CX,4H  CY,4H  CZ,4HL   ,4HM   ,4HN   ,
     44H  FA,4HX   ,4HY   ,4HZ   ,4H  TA,4H  VN,4H  VE,4H  VD,4H  VR,
     54HW   ,4HQ   ,4H  XM,4HACH ,4H  XC,4HG   ,4H  YC,4H  HC,4H  XI,
     64HXX  ,4HYY  ,4HZZ  ,4HXZ  ,4H  WA,4HIT  ,4H  CG,4H  SP,4HAN  ,
     74H  CH,4HORD ,4H  AR,4HEA  ,4H  DE,4H  DA,4H  DR,4H    ,4H  FE,
     84H  FG,4H  FT,4H  TE,4H  TG,4H  TT,4HP1  ,4HP2  ,4HP3  ,4HP4  ,
     94H  T1 ,4H  T2,4H  T3,4H  T4,4HO   ,4H  DX,4H  DY,4H  DH,
     A 4HQ   ,4HPK  ,4H   ,4H   ,4H    /
C
      VPWK=VPW/1.69
      GAMHD=GAMH*R2D
      GAMVD=GAMV*R2D
```

```
      CALL DATE(DMY)
      CALL TIME(HMS)
      PRINT 1,TITLE
C
      PRINT 1000 ,N2,DMY,HMS
C     CONVERT      ALL VARIABLES TO METRIC UNITS FOR IC PRINTOUT
      EMLBKG=.45359
      EMINF=1.3556
      EMMAS=14.5919
      EMFORC=4.418
      EMTORQ=1.3546
      EMFTM=.3048
      EMICM=2.54
      EMAREA=.092903
      EMKMS=.51479
      EMMPHMS=.447039
      EMDPFS=47.8113
      XIXXM=XIXX*EMINF
      XIYYM=XIYY*EMINF
      XIZZM=XIZZ*EMINF
      XIXZM=XIXZ*EMINF
      WAITM=WAIT*EMLBKG
      SPANM=SPAN*EMFTM
      CHORDM=CHORD*EMFTM
      AREAM=AREA*EMAREA
      XCGM=XCG*EMFTM
      YCGM=YCG*EMFTM
      HCGM=HCG*EMFTM
      FAXM=FAX*EMFORC
      FAYM=FAY*EMFORC
      FAZM=FAZ*EMFORC
      FEXM=FEX*EMFORC
      FEYM=FEY*EMFORC
      FEZM=FEZ*EMFORC
      FGXM=FGX*EMFORC
      FGYM=FGY*EMFORC
      FGZM=FGZ*EMFORC
      FTXM=FTX*EMFORC
      FTYM=FTY*EMFORC
      FTZM=FTZ*EMFORC
      TALM=TAL*EMTORQ
      TAMM=TAM*EMTORQ
      TANM=TAN*EMTORQ
      TELM=TEL*EMTORQ
      TEMM=TEM*EMTORQ
      TENM=TEN*EMTORQ
      TGLM=TGL*EMTORQ
      TGMM=TGM*EMTORQ
      TGNM=TGN*EMTORQ
      TTLM=TTL*EMTORQ
      TTMM=TTM*EMTORQ
      TTNM=TTN*EMTORQ
      VNM=VN*EMFTM
      VEM=VE*EMFTM
      VDM=VD*EMFTM
```

```
      VNDM=VND*EMFTM
      VEDM=VED*EMFTM
      VDDM=VDD*EMFTM
      VRWM=VRW*EMFTM
      PRINT 2,QDAT(45),QDAT(46),XIXXM,UDAT(9),QDAT(45),QDAT(47),XIYYM,
000001 UDAT(9),QDAT(45),QDAT(48),XIZZM,UDAT(9)
C
       PRINT 2,QDAT(45),QDAT(49),XIXZM,UDAT(9),QDAT(50),QDAT(51),WAITM,
000001 UDAT(13),QDAT(52),QDAT(16),CG
C
       PRINT 2,QDAT(53),QDAT(54),SPANM,UDAT(8),QDAT(55),QDAT(56),CHORDM
000001,UDAT(8),UDAT(57),QDAT(58),AREAM,UDAT(10)
      PRINT 3
C
      PRINT 2, NQDAT(1),NQDAT(2),A1,U(1)
000001,        NQDAT(3),NQDAT(4),A2,U(2)
000002,        NQDAT(5),NQDAT(6),A3,U(3)
C
      PRINT 2, NQDAT(7),NQDAT(8),A4,U(4)
000001,        NQDAT(9),NQDAT(10),A5,U(5)
000002,        NQDAT(11),NQDAT(12),A6,U(6)
C
      PRINT 2, NQDAT(13),NQDAT(14),A7,U(7)
000001,        NQDAT(15),NQDAT(16),A8,U(8)
000002,        NQDAT(17),NQDAT(18),A9,U(9)
      PRINT 3
C
C
C
C
       PRINT 2,UDAT(78),QDAT(81),XCGM,UDAT(8),QDAT(79),QDAT(81),YCGM
000001,UDAT(8),QDAT(80),QDAT(81),HCGM,UDAT(8)
      PRINT 2,QDAT(1),QDAT(2),PHI,UDAT(1),QDAT(3),QDAT(4),THET
000001,UDAT(1),QDAT(5),QDAT(2),PSI,UDAT(1)
C
      PRINT 2,QDAT(6),QDAT(7),ALFA,UDAT(1),QDAT(8),QDAT(9),BETA
000001,UDAT(1),QDAT(10),QDAT(11),GAMVU ,UDAT(1)
C
      PRINT 2,UDAT(6),QDAT(12),ALFD,UDAT(2),QDAT(8),QDAT(13),BETD
000001,UDAT(2),UDAT(10),QDAT(14),GAMMD ,UDAT(1)
      PRINT 3
C
      PRINT 2,UDAT(15),QDAT(16),PR,UDAT(2),QDAT(17),QDAT(16),QB
000001,UDAT(2),QDAT(18),QDAT(16),RB,UDAT(2)
C
      PRINT 2,QDAT(15),QDAT(19),PRD,UDAT(3),QDAT(17),QDAT(19),QBD
000001,UDAT(3),UDAT(18),QDAT(19),RBD,UDAT(3)
      PRINT 3
C
C
       PRINT 2,UDAT(20),QDAT(16),CD,UDAT(14),QDAT(21),QDAT(16),CL
       PRINT 2,QDAT(22),QDAT(16),CX,UDAT(14),QDAT(23),QDAT(16),CY
000001,UDAT(14),QDAT(24),QDAT(16),CZ
C
      PRINT 2,QDAT(21),QDAT(25),CLL,UDAT(14),QDAT(21),QDAT(26),CLM
```

```
000001 ,UDAT(14),QDAT(21),QDAT(27),CLN
      PRINT 3
C
       PRINT 2,QDAT(24),QDAT(29),FAXM,UDAT(4),QDAT(28),QDAT(30),FAYM
000001,UDAT(4),QDAT(28),QDAT(31),FAZM,UDAT(4)
       PRINT 2,QDAT(63),QDAT(29),FEXM,UDAT(4),QDAT(63),QDAT(30),FEYM
000001,UDAT(4),QDAT(63),QDAT(31),FEZM,UDAT(4)
C
       PRINT 2,QDAT(64),QDAT(29),FGXM,UDAT(4),QDAT(64),QDAT(30),FGYM
000001,UDAT(4),QDAT(64),QDAT(31),FGZM,UDAT(4)
       PRINT 2,QDAT(65),QDAT(29),FTXM,UDAT(4),QDAT(65),QDAT(30),FTYM
000001,UDAT(4),QDAT(65),QDAT(31),FTZM,UDAT(4)
      PRINT 3
C
       PRINT 2,QDAT(32),QDAT(25),TALM,UDAT(5),QDAT(32),QDAT(26),TAMM
000001,UDAT(5),QDAT(32),QDAT(27),TANM,UDAT(5)
       PRINT 2,QDAT(66),QDAT(25),TELM,UDAT(5),QDAT(66),QDAT(26),TEMM
000001,UDAT(5),QDAT(66),QDAT(27),TENM,UDAT(5)
C
       PRINT 2,QDAT(67),QDAT(25),TGLM,UDAT(5),QDAT(67),QDAT(26),TGMM
000001,UDAT(5),QDAT(67),QDAT(27),TGNM,UDAT(5)
C
       PRINT 2,QDAT(68),QDAT(25),TTLM,UDAT(5),QDAT(68),QDAT(26),TTMM
000001,UDAT(5),QDAT(68),QDAT(27),TTNM,UDAT(5)
      PRINT 3
C
       PRINT 2,QDAT(33),QDAT(19),VNDM,UDAT(7),QDAT(34),QDAT(19),VEDM
000001, UDAT(7),QDAT(35),QDAT(19),VDDM,UDAT(7)
       PRINT 2,QDAT(36),QDAT(37),VRWM,UDAT(6),QDAT(34),QDAT(38),VEQ
000001, UDAT(12),QDAT(39),QDAT(40),XMACH
       PRINT 2,QDAT(36),QDAT(82),VPWK,UDAT(12)
      PRINT 3
C
C
C
C
C
C
      RETURN
    1 FORMAT(1H1,35X,A4,27H INITIAL CONDITION PRINTOUT/        )
    2 FORMAT(10X,2A4,1H=,E11.4,1X,A4,5X,2A4,1H=,E11.4,1X,A4,5X
000001,          2A4,1H=,E11.4,1X,A4)
00003 FORMAT(1H0)
01000 FORMAT(10X,10HRUN NUMBER,4X,I3,15X,5HDATE:,3X,A9,14X,5HTIME
000002  ,3X,A10/)
      END
```

••

```
      SUBROUTINE WINDC
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
       COMMON/CHF/H(200)/ICHF/IH(50)
       COMMON/GAIN1/GA(100)
```

```
      EQUIVALENCE(T11,A(16)),(T21,A(17)),(T31,A(18))
     1 ,(T12,A(19)),(T22,A(20)),(T32,A(21))
     2 ,  (T13,A(22)),(T23,A(23)),(T33,A(24))
      EQUIVALENCE(VNTURB,A(419)),(VETURB,A(420)),(VDTURB,A(421))
      EQUIVALENCE   (A(61),UTURB)
      EQUIVALENCE(A(62),VTURB)
      EQUIVALENCE(A(63),WTURB)
      EQUIVALENCE(A(70),VRW)
      EQUIVALENCE(A(168),DT)
      EQUIVALENCE(A(238),VEQIC)
       EQUIVALENCE(XCB,A(174)),(YCB,A(175)),(STHETR,A(114)),(CTHETR,
     1  A(115))
       EQUIVALENCE(ALT,A(83))
      EQUIVALENCE(A(303),TIME)
      EQUIVALENCE(IA(1),IMODE)
       EQUIVALENCE(IONCE,IB(16))
       EQUIVALENCE(VS,B(30))
       EQUIVALENCE(PHIP,B(112))
       EQUIVALENCE(OMEAP,B(111))
       EQUIVALENCE(UGST,B(153)),(VGST,B(154)),(WGST,B(155))
       EQUIVALENCE(XFRZ,B(156)),(SWBSW,B(157)),(SUBSW,B(158))
       EQUIVALENCE(IB(18),IPITCH)
       EQUIVALENCE(PITCHM,B(110))
       EQUIVALENCE(NCHK,IB(22))
      EQUIVALENCE(B(95),X2S),(B(94),X1S)
       EQUIVALENCE(ILWURB,IB(25))
       EQUIVALENCE(IHFRZ,IB(24)),(IWIND,IB(20)),(ILTURB,IB(21))
       EQUIVALENCE(GA(30),GR(1))
      EQUIVALENCE(HIS,B(100))
      EQUIVALENCE(IUISCT,IA(50))
      EQUIVALENCE(FRTN,B(194)),(FRTE,B(195)),(FRTD,B(196)),(VMN,B(197))
      EQUIVALENCE(VME,B(198)),(VMD,B(199)),(TDTURB,B(200))
      DIMENSION RD(7),GR(7)
       DIMENSION RDY(7),RDYI(7),IRDST(7),IRDSI(7,10)
       DIMENSION RDX(7)
       DATA SWBSW/1./,SUBSW/1./
       DATA ISET/0/
      DATA TIM3/0./
       DATA(GR(I),I=1,7)/6.,6.,.125,.125,.25,.3,.3./
      DATA RD6/.1/
      DATA PI/3.141592/
      DATA FC/5./
       DATA (IRDST(I),  I=1,7)/1,3,5,7,9,11,13/
      DATA IUMT/0/,IVAT/0/,IWBT/0/
       DATA ZR1/.001/,ZR2/1.5/,YR1/.001/,YR2/1.5/
       DATA XFRZ/1500./
       DATA((IRDSI(I,J),I=1,7),J=1,10)/1,3,5,7,9,11,13,13,1,3,5,7,9,11,
     1  11,13,1,3,5,7,9,9,11,13,1,3,5,7,7,9,11,13,1,3,   5,5,7,9,11,
     1  13,1,3,3,  5,7,9,11,13,1,15,17,19,21,23,25,27,27,15,17,19,21,
     3  23,25,25,27,15,17,19,21,23/
       IF(IUISCT)7,7,30
 7     IF(IMODE)2,3,4
 2     CONTINUE
       DT2=DT
      VO=VEQIC*1.689
```

```
      IF(VO.LT.60)VO=60.
      RK10=SQRT(3.14159*5900.)*.003
      RK11=SQRT(5900.*3.14159)*3.*VO/400000.
      S110=-VO/1000.
      S210=-3.*VO/400.
      Z01=EXP(S110*DT)
      Z02=EXP(S210*DT)
      Z11=-1./(S110-S210)*RK10*(1.-Z01)
      Z12=1./(S110-S210)*RK10*(1.-Z02)
      Z13=-1./(S210-S110)*RK11*(1.-Z01)/S110
      Z14=1./(S210-S110)*RK11*(1.-Z02)/S210
      Z0111  =EXP(-VO/100.*DT2)
      Z1111=SQRT(71.63*3.14159)/100.*(1.-Z0111)
      Z0112=EXP(-VO/100.*DT2)
      Z1112=SQRT(200.*3.14159)/100.*(1.-Z0112)
      IX=2**24-3
      DO 10 I=1,7
      RDX(I)=0.
      RDYI(I)=0.
      IF(IONCE.GT.0)  IRDST(I)=IRDSI(I,IONCE)
00010 CONTINUE
      RN1=.044*SQRT(PI)
      WC=2.*FC*PI
      IF(VEQIC.NE.VEQICP)  IZX=1
      VEQICP=VEQIC
    3 CONTINUE
      WASS=0.
      UASS=0.
      UA2S=0.
      RDSP=0.
      RD4=0.
      RD3=0.
      UG=0.
      WG=0.
      WA2S=0.
      UA2SS=0.
      VRSA=0.
      ZRSA=0.
      UGSA=0.
      WGSA=0.
      WGSA=0.
      UA2TA=0.
      WA2TA=0.
      WA2SS=0.
      UGS=0.
      VG1=0.
      VG2=0.
      VG3=0.
      VG4=0.
      VGS=0.
      WGS=0.
      UGSS=0.
      VGSS=0.
      WGSS=0.
      ZRS=0.
```

```
      YRS=0.
      ZRSS=0.
      YRSS=0.
      FACY=EXP(-ABS(YCG)/100.)
      ALT1=H1S+50.-XCG*.1
      FACZ=1.
      IF(ALT.GT.ALT1) FACZ=EXP((ALT1-ALT)/50.)
      FACW=FACY*FACZ
      XD=XCG
      UTURB=-VS*SUBSW*FACW*TAB1(XD,0.,0.,68,1)
      WTURB=-VS*SWBSW*FACW*TAB1(XD,0.,0.,69,1)
      VTURB=0.
      TIM3=0.
      ZQ=0.
      YQ=0.
      RETURN
6     CONTINUE
      IF(I=IND.EQ.0)RETURN
       IF(NCHK.GT.0) RETURN
      FACY=EXP(-ABS(YCG)/100.)
      ALT1=H1S+50.-XCG*.1
       FACZ=1.
      IF(ALT.GT.ALT1) FACZ=EXP((ALT1-ALT)/50.)
       FACW=FACY*FACZ
      XD=XCG
       IF(IBFRZ.EQ.1) XD=-XFRZ
       UHSS1=-VS*TAB1(XD,0.,0.,68,1)
       UBSS=UHSS1*SUBSW*FACW
       IF(ILBURB)21,21,20
00021  WBSS1=-VS*TAB1(XD,0.,0.,69,1)
       WBSS=WBSS1*SWBSW*FACW
       GO TO 22
00020  WBSS1=-VS*TAB1(XD,0.,0.,70,1)
       WBSS=WBSS1*SWBSW*FACW
00022  CONTINUE
      VSD=1./(.95*VS)
       CTERM=COS(OMEGP*((1.+(VBW-VS)*VSD)*TIME+XD*VSD+PHIP-1.57))*IPITCH
      UB1K=2.22+.0004*XD
      IF(UB1K.LT.0.0) UB1K=0.0
       UB1=PITCHM*VS*UB1K*CTERM*FACW
       IF(XCG.LT.-2236.) UB1=0.
      IF(XCG.GT.X2S) UB1=0.
      WB1K=4.90+.0018*XD
      IF(WB1K.LE.0.0) WB1K=0.0
       WB1=PITCHM*VS*WB1K*CTERM*FACW
       IF(XCG.LT.-2536.) WB1=0.
      IF(XCG.GT.X2S) WB1=0.
      SINWCT=SIN(WC*TIME)
       DO 1 I=1,7
       CALL RAND(IRDST(I),IX,1,1,U)
       RDX(I)=U
      RDY(I)=RDX(I)-RDYI(I)
      RDYI(I)=RDYI(I)+RDA*RDY(I)*DT
      RD(I)=RDY(I)*SINWCT
1     CONTINUE
```

```
 SUB2=TAB1(XD,0.,0.,71,1)
 TUB2=TAB1(XD,0.,0.,72,1)
DTOT=DT/TUB2
 UA2=UB2P*(1.-DTOT)+VS*SUB2*1.414*SQRT(1./TUB2)*DTOT*RD(1)*GR(1)
UA2P=UA2
DTOTW=DT/3.33
 WA2=WA2P*(1.-DTOTW)+.035*VS*.748*DTOTW*RD(2)*GR(2)
 IF(XD.LT.-2540.) WA2=0.
IF(XD.GT.X1S) WA2=0.
WA2P=WA2
 VYAR=WA2
 VXAR=UBSS+UA1+UA2
 VDAR=WBSS+WA1+WA2
 VNAR=VXAR*CTHETR-VYAR*STHETR
VEAR=VYAR*CTHETR+VXAR*STHETR
 UAT=T11*VNAR+T12*VEAR+T13*VDAR
 VAT=T21*VNAR+T22*VEAR+T23*VDAR
 WAT=T31*VNAR+T32*VEAR+T33*VDAR
IF(IUWT.EQ.1) UWT=0.
IF(IVWT.EQ.1) VWT=0.
IF(IWWT.EQ.1)WWT=0.
 WG=Z0111*WG+Z1111*RD3
RD3=RD(3)*GR(3)
 UG=Z0112*UG+Z1112*RD4
RD4=RD(4)*GR(4)
 VG1=Z01*VG1+Z11*RD5P
VG2=Z02*VG2+Z12*RD5P
 VG3=Z01*VG3+Z13*RD5P
 VG4=Z02*VG4+Z14*RD5P
 RD5P=RD(5)*GR(5)
 VG=VG1+VG2+VG3+VG4
 IF(ILTURB.EQ.1) GO TO 5
UG=0.0
VG=0.
WG=0.
5     CONTINUE
UTURB=UWT+UG
VTURB=VWT+VG
WTURB=WWT+WG
 ZR=WN1*RD(6)*GR(6)*(ZR1*ABS(XD)+ZR2)
 YR=WN1*RD(7)*GR(4)*(YR1*ABS(XD)+YR2)
UA2S=UA2S+UA2
WA2S=WA2S+WA2
UA2SS=UA2SS+DT*UA2**2
WA2SS=WA2SS+DT*WA2**2
UGS=UGS+UG
VGS=VGS+VG
WGS=WGS+WG
UGSS=UGSS+DT*UG**2
VGSS=VGSS+DT*VG**2
WGSS=WGSS+DT*WG**2
ZRS=ZRS+ZR
YRS=YRS+YR
ZRSS=ZRSS+DT*ZR**2
YRSS=YRSS+DT*YR**2
```

```
      IF(TIME)  30,30,25
00025 CONTINUE
      RAV=TIME/DT
      YRSA=YRS/RAV
      ZRSA=ZRS/RAV
      UGSA=UGS/RAV
      VGSA=VGS/RAV
      WGSA=WGS/RAV
      UH2TA=UH2S/RAV
      WH2TA=WH2S/RAV
      UH2ST=UH2SS/TIME
      WH2ST=WH2SS/TIME
      UGST=UGSS/TIME
      VGST=VGSS/TIME
      WGST=WGSS/TIME
      ZRST=ZRSS/TIME
      YRST=YRSS/TIME
00030 CONTINUE
      IF(IDISCT)37,37,33
 33   CONTINUE
      FRT=AMAX1(FRTN,FRTE,FRTD)
      PHASE=FRT*(TIME-TDTURB)
      IF(PHASE-6.28)34,34,36
 34   IF(TIME-TDTURB)36,36,35
 35   VNTURB=.5*VMN*(1.-COS(FRTN*(TIME-TDTURB))      )
      VETURB=.5*VME*(1.-COS(FRTE*(TIME-TDTURB))   )
      VDTURB=.5*VMD*(1.-COS(FRTD*(TIME-TDTURB)) )
      GO TO 37
 36   VNTURB=0.
      VETURB=0.
      VDTURB=0.
 37   CONTINUE
      RETURN
      END
```

****

```
      SUBROUTINE RAND(IU,IX,K,N,U)
      DIMENSION U(N)
      IF(K)11,7,11
 7    DO 8 I=1,7
      IU=IU*IX
 8    U(I)=IU/281474976710655.
      RETURN
 11   DO 13 I=1,N
      S=0.
      DO 12 J=1,12
      IU=IU*IX
 12   S=S+IU
 13   U(I)=S/281474976710655.-6.
      RETURN
      END
```

```
      SUBROUTINE CONTR2
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
      COMMON/CHF/B(200)/ICBF/IB(50)
      EQUIVALENCE(AIL,B(2))
      EQUIVALENCE(RUD,B(3))
      EQUIVALENCE(STAB1,B(5))
      EQUIVALENCE(PSAS,B(6))
      EQUIVALENCE(RLONSTK,B(7))
      EQUIVALENCE(RLATSTK,B(9))
      EQUIVALENCE(RUDPED,B(11))
      EQUIVALENCE(YSAS,B(12))
      EQUIVALENCE(RSAS,B(13))
      EQUIVALENCE(A(37),PB)
      EQUIVALENCE(QB,A(38))
      EQUIVALENCE(RB,A(39))
      EQUIVALENCE(AY,A(92))
      EQUIVALENCE(IMODE,IA(1))
      EQUIVALENCE(IPDAMP,IB(26))
      EQUIVALENCE(IQDAMP,IB(27))
      EQUIVALENCE(IRDAMP,IB(28))
      EQUIVALENCE(DT,A(168))
      EQUIVALENCE(RKQ,B(160))
      EQUIVALENCE(TLEADP,B(161))
      EQUIVALENCE(PSASI,B(162))
      EQUIVALENCE(PSASII,B(163))
      EQUIVALENCE(PHWSHI,B(165))
      EQUIVALENCE(PHWSH,B(166))
      EQUIVALENCE(RKDRRB,B(167))
      EQUIVALENCE(TLAGRB,B(168))
      EQUIVALENCE(RKDRAY,B(169))
      EQUIVALENCE(TLEADAY,B(170))
      EQUIVALENCE(TLAGAY,B(171))
      EQUIVALENCE(AYFILTI,B(172))
      EQUIVALENCE(STAB,B(1))
      EQUIVALENCE(AYFILT,B(173)),(YSAS2,B(174)),(RINCNCT,B(175)),
     1(RKDRDA,B(176)),(B(177),YSAS1),(YSASLIM,B(178)),(RPHDA,B(179)),
     2 (TLEADDA,B(180)),(TLAGDA,B(181)),(PHWSH,B(182)),(PWSHI,B(183)),
     3 (PSASLIM,B(184)),(AILC,B(185)),(TAUACDA,B(186)),(TAUACHS,
     4B(187)),(RUD2,B(188))
      EQUIVALENCE(IA(16),I2DOF),(TIME,A(303))
      DATA RKQ/28.363/,TLEADP/.169/,TLAGP/.741/,PSASLIM/1.5/
      DATA RKDRDA/.58/,RPHDA/19.825/,TLEADDA/.781/
      DATA TLAGDA/1.29/
      DATA TAUACHS/.0833/,TLAGRB/3.11/,RKDRRB/38.33/,RKDRAY/.83/
      DATA TLEADAY/.25/,TLAGAY/.125/,YSASLIM/5./,RSASLIM/2./
      DATA TAUACDA/.04/,IPDAMP/1/,IRDAMP/1/,IQDAMP/1/
      IF(IMODE)1,100,2
00001 PSASII=0.
      PWSHI=0.
      PHWSHI=0.
      AYFILTI=0.
      PWSHI=0.
      EXPDTHS=1.-EXP(-DT/TAUACHS)
      EXPDTDA=1.-EXP(-DT/TAUACDA)
      AIL=AILC
```

```
      STAB=STABC
00002 CONTINUE
      STAB1=TAB1(RLONSTK,0.,0.,1,1)
      PSAS1I=PSAS1I+(RKQ*QB-PSAS1)*DT
      PSAS1=(PSAS1I+RKQ*TLEADP*QB)/TLAGP
      PSAS1=PSAS1*IQUAMP
      PSAS=PSAS1
      IF(ABS(PSAS1).GT.PSASLIM)PSAS=SIGN(PSASLIM,PSAS1)
      STABC=STAB1+PSAS
      STAB=STAB+EXPDTHS*(STABC-STAB)
      IF(IZUUF.EQ.1)STAB=TAB1(TIME,0.,0.,74,1)
C   YAW SAS
      RBWSHI=RBWSHI+RBWSH*DT
      RBWSH=(RKDRRA*RB-RBWSHI)/TLAGRB
      AYFILTI=AYFILTI+(RKDRAY*AY-AYFILT)*DT
      AYFILT=(AYFILTI+RKDRAY*TLEADAY*AY)/TLAGAY
      PWSHI=PWSHI+(RPHDA*PB-PWSH)*DT
      PWSH=(PWSHI+RPHDA*TLEADDA*PB)/TLAGDA
      PBWSH=PWSH
      IF(ABS(PWSH).GT.RSASLIM)PBWSH=SIGN(RSASLIM,PBWSH)
      RSAS=-PBWSH*IPDAMP
      AILC=TAB1(PLATSTK,0.,0.,3,1)
      AIL=AIL+EXPDTDA*(AILC+RSAS-AIL)
      RINCNCT=AILC*RKDRDA
      YSAS2=(RBWSH+AYFILT)*IRDAMP
      YSAS1=YSAS2-RINCNCT
      YSAS=YSAS1
      IF(ABS(YSAS1).GT.YSASLIM) YSAS=SIGN(YSASLIM,YSAS1)
C    ROLL SAS
      RUD=TAB1(RUDPED,0.,0.,2,1)
      RUD1=RUD
      RUD2=RUD1+YSAS
00100 CONTINUE
      IF(IMODE)101,101,102
00101 STAB=STABC
      AIL=AILC
00102 CONTINUE
      RETURN
      END
```

```
**********************************************************************

      SUBROUTINE AERO2
      COMMON/CHF/B(200)/ICHF/IB(50)
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
      EQUIVALENCE(ALT,A(83))
      EQUIVALENCE(STAB,B(1)),(RUD,B(3)),(AIL,B(2)),(THTN,B(17))
      EQUIVALENCE(TIME,A(303)),(IMODE,IA(1)),(VJR,B(122)),
000001(DXCG,A(198)),(DZCG,A(271)),(CHORD,A(182)),(SPAN,A(181)),
000002(ALFA,A(180)),(QBAR,A(178)),(RMACH,A(71)),(VRW,A(70))
      EQUIVALENCE(RHO,A(183)),(CDQ,B(123)),(CLADOT,B(124)),(CLP,B(125))
      EQUIVALENCE(CMQ,B(126)),(CYDR,B(127)),(CYR,B(128)),(CLDA,B(129)),
000001(CLQ,B(130)),(CMADT,B(131)),(CMDR,B(132)),(CNR,B(133)),(CYDA,
```

```
000002H(134)),(CYP,H(135)),(DXCGO,A(254)),(DZCGO,A(255))
      EQUIVALENCE(PHWN,A(49)),(QBWN,A(50)),(RBWN,A(51))
      EQUIVALENCE(PTURB,A(52)),(QTURB,A(53)),(RTURB,A(54))
      EQUIVALENCE(A(25),ALFA),(A(26),BETA),(A(29),ALFD),(A(31),SALPH),
000001(A(32),CALPH),(A(37),PH),(A(38),QB),(A(39),RB),(A(155),TAL),
000002 (A(156),TAM),(A(157),TAN)
      EQUIVALENCE(VEG,A(75))
      EQUIVALENCE(TEN,B(58)),(TRN,B(59))
      EQUIVALENCE(FAX,A(136)),(FAY,A(137)),(FAZ,A(138))
      EQUIVALENCE(CL,A(131)),(CD,A(132)),(CX,A(133)),(CY,A(134))
      EQUIVALENCE(CZ,A(135)),(CLL,A(152)),(CLM,A(153)),(CLN,A(154))
      EQUIVALENCE(B(151),CMSTR1),(CLSTR3,B(152))
      EQUIVALENCE(ISTOR1,IB(37)),(ISTOR2,IB(38))
      DATA CMSTR1/.05/,CDSTR1/.0052/
      DATA CLSTR3/0./
      DATA ISTOR1/1/,ISTOR2/1/
      DATA CLSTR2/0./,CMSTR2/0./,CDSTR2/.01395/
      DATA CDO/0./,CLADOT/0./,CLP/-.3152/,CMQ/-10.49/,
000001CYDR/.00165/,CYR/.4469/,CLDA/.0017/,CLQ/4.87/,
000002CMADT/-10./,CNDR/-.0011/,CNR/-.45/,CYDA/0./,
000003CYP/.07/,DXCGO/28.5/,DZCGO/8.015/
      IF(QBAR)1,1,2
    1 VJR=11.9
      VER1=1000.
      GO TO 3
    2 VJR=SQRT((TEN+TRN)/(QBAR*13.))
    3 CONTINUE
      IF(VJR.GT.0.) VER1=VJR
      VER=1./VER1
      IF(THTN.LE.30..AND.VJR.GE.5.4) VJR=5.4
      IF(VJR.LT .5.4) VJR=5.4
      IF(VJR.GT.11.9)VJR=11.9
      GF=QBAR*AREA
      PHWN=PH+PTURB
      RBWN=RB+RTURB
      QBWN=QB+QTURB
      PS=PHWN*CALPH+RBWN*SALPH
      RS=RBWN*CALPH-PHWN*SALPH
      IF(VRW-1.)4,4,5
    4 CO2V=0.
      BO2V=0.
      GO TO 6
    5 CONTINUE
      CO2V=CHORD/(2.*VRW)
      BO2V=SPAN/(2.*VRW)
    6 CONTINUE
      CLSTR1=TAB1(VER,THTN,0.,73,1)
      CLTANK=CLSTR3
      IF(THTN.LE.50.) CLTANK=CLSTR3*THTN/50.
      CLSTORE=(CLSTR1+CLTANK)*ISTOR1+CLSTR2*ISTOR2
      CDSTORE=CDSTR1*ISTOR1+CDSTR2*ISTOR2
      CMTANK=CMSTR1
      IF(THTN.LE.45.) CMTANK=CMSTR1*THTN/45.
      CMSTORE=CMTANK*ISTOR1+CMSTR2*ISTOR2
      CL1=TAB1(ALFA,STAB,THTN,37,6)
```

```
      CL2=TAB1(ALFA,STAB,THTN,43,3)
      CL3=CL1+(CL2-CL1)*(VJR-5.4)/6.5
      CL=CL3+(CLADOT*ALFD+CLQ*QBWN)*CO2V+CLSTORE
      CD1=TAB1(ALFA,STAB,THTN,46,6)
      CD2=TAB1(ALFA,STAB,THTN,52,3)
      CD3=CD1+(CD2-CD1)*(VJR-5.4)/6.5
      CD=CD3+(CDQ*QBWN)*CO2V+CDSTORE
      CX=-CD*CALPH+CL*SALPH
      CZ=-CD*SALPH-CL*CALPH
      CM1=TAB1(ALFA,STAB,THTN,55,6)
      CM2=TAB1(ALFA,STAB,THTN,61,3)
      CM3=CM1+(CM2-CM1)*(VJR-5.4)/6.5   +CMSTORE
      CLM=CM3+CO2V*(CMADOT*ALFD+CMQ*QBWN)
000001 -CZ*(DXCG-DXCGO)/CHORD+CX*(DZCG-DZCGO)/CHORD
      QSB=QF*SPAN
      QSCB=QF*CHORD
      FAX=QF*CX
      FAZ=QF*CZ
      CY1=TAB1(BETA,ALFA,THTN,19,3)
      CY2=TAB1(BETA,ALFA,THTN,22,2)
      CY3=CY1+(CY2-CY1)*(VJR-5.4)/6.5
      CY=CY3+CYDR*RUD+CYDA*AIL+(CYP*PS+CYR*RS)*BO2V
      CLL1=TAB1(BETA,ALFA,THTN,24,3)
      CLL2=TAB1(BETA,ALFA,THTN,27,2)
      CLL3=CLL1+(CLL2-CLL1)*(VJR-5.4)/6.5
      CLDR=TAB1(VEQ,0.,0.,34,1)
      CLR=TAB1(ALFA,0.,0.,36,1)
      CLLS=CLL3+CLDR*RUD+CLDA*AIL+(CLR*RS+CLP*PS)*BO2V
      CN1=TAB1(BETA,ALFA,THTN,29,3)
      CN2=TAB1(BETA,ALFA,THTN,32,2)
      CN3=CN1+(CN2-CN1)*(VJR-5.4)/6.5
      CNDA=TAB1(ALFA,0.,0.,35,1)
      CNP=TAB1(ALFA,0.,0.,66,1)
      CNS=CN3+CNDA*AIL+CNDR*RUD+(CNP*PS+CNR*RS)*BO2V
      CLL=CLLS*CALPH-CNS*SALPH
      CLN=CLLS*SALPH+CNS*CALPH
      TAL=QSB*CLL
      TAM=QSC*CLM
      TAN=QSB*CLN
      FAY=QF*CY
      RETURN
      END
```

****

```
      SUBROUTINE GEARS
      COMMON/CHF/B(200)/ICHF/IB(50)
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
      DIMENSION DYPA(15),DXPA(15),DZPA(15)
      DIMENSION HDTOKE(7),HDECK(5),HIG(6),ITOUCH(5)
      EQUIVALENCE(FAZ,A(138)),(FEZ,A(141))
      EQUIVALENCE(HDTOKE(1),A(447)),(HIG(1),A(291)),(HDECK(1),A(272))
      EQUIVALENCE(RLAM1,B(80)),(B(81),RLAM2),(DT,A(160)),(IMODE,IA(1))
```

```
      EQUIVALENCE(FEX,A(139)),(FAX,A(136)),(WAIT,A(177))
      EQUIVALENCE(PB,A(37)),(ITOUCH(1),IA(10))
      EQUIVALENCE(DYPA(1),A(210)),(DXPA(1),A(184)),(DZPA(1),A(257))
      EQUIVALENCE(FGXM,A(455))
      EQUIVALENCE(ALT,A(83)),(PHIR,A(4)),(PSIR,A(6)),(THETR,A(5))
      EQUIVALENCE(UNR,A(106)),(DER,A(107))
      EQUIVALENCE(VRW,A(70))
      EQUIVALENCE(VR,A(454))
      EQUIVALENCE(VX,A(200))
      EQUIVALENCE(FGX,A(142)),(FGY,A(143)),(VN,A(64)),(VE,A(65)),
     2 (CPSI,A(15)),(SPSI,A(14)),(FGZ,A(144)),(TGM,A(162)),
     3 (TGL,A(161)),(TGN,A(163)),(PHIR,A(4)),(THETR,A(5)),(ALT,A(83)),
     4 (ALTD,A(80)),(PSIR,A(6))
      EQUIVALENCE(IBRAK,IB(29)   )
      EQUIVALENCE(IB(9),ICRASH)
      EQUIVALENCE(SPHI,A(10)),(CPHI,A(11)),(STHT,A(12)),(CTHT,A(13))
      EQUIVALENCE(IMG,       IA(11))
      EQUIVALENCE(QB,      A(38))
      EQUIVALENCE(ALT,A(83)),(ALTD,A(80))
      DATA RMUX/.6/
      RMUX=.6
      IF(IBRAK.EQ.0)RMUX=.03
      DZNG=HIG(1)-HDECK(2)
      DZMG=HIG(2)-HDECK(3)
      IF(IMODE)1,1,2
00001 DZNGP=DZNG
      DZMGP=DZMG
      DZMG1=DZMG
      IF(DZMG1.GT.0.) DZMG1=0.
      DYGEAR=DYPA(3)-DYPA(4)
      DXGEAR=DXPA(1)-DXPA(2)
00002 CONTINUE
      RLAM1U=(HDTDKE(2)-HDTDKE(3))/DXGEAR
      RLAM2U=(HDTDKE(5)-HDTDKE(4))/DYGEAR
      FGZNG=TAB1(DZNG,0.,0.,64,1)+ITOUCH(1)*((DZNG-DZNGP)/DT*2500.
     1 +250.*DZNG)
      FGZMG=TAB1(DZMG,0.,0.,65,1)+ITOUCH(2)*((DZMG-DZMGP)/DT*2500.+
     1 250.*DZMG)
      DZNGP=DZNG
      DZMGP=DZMG
      FGZ=FGZNG+FGZMG
      IF(FGZ.GT.0.) FGZ=0.
      IF(VR.NE.0.) FGX=(RMUX*FGZMG+.03*FGZNG)*SIGN(1.,VR)
      IF(ABS(VR).LT..01) FGX=(-(FAX+FEX)*COS(THETR-RLAM1)+SIN(RLAM1)*
     1 WAIT*   (FAZ+FEZ)*SIN(THETR-RLAM1))*ITOUCH(2)
      FGXM=-(RMUX*FGZMG+.03*FGZNG)
      IF(ABS(FGX).GT.FGXM)FGX=SIGN(FGXM,FGX)
      TGM=-FGZNG*(DXPA(1))-FGZMG*(DXPA(2))+FGX*(DZPA(2)+DZMG1)
      IF(ITOUCH(3).EQ.1.AND.ITOUCH(4).EQ.1) GO TO 3
      GO TO 4
00003 RLAM=RLAM2U
      PHIG=RLAM2
00004 CONTINUE
      IF(IMODE.EQ.-1) GO TO 20
      IF(DZNG.LT.-1..AND.DZMG.LT.-1.) GO TO 7
```

```
      GO TO 8
00007 ALTD=HOTDKF(1)
      QH=PLAM1D
      Z1=DZPA(1)+DZNG
      Z2=DZPA(2)+DZNG
      Z3=Z1-Z2
      X1=DXPA(1)-DXPA(2)
      THET1=ATAN2(Z3,X1)
      THETH=THET1+RLAM1
      GO TO 20
00008 IF(DZNG.LT.-1.AND.QB.LT.0.) QB=0.
      IF(DZNG.LT.-1..AND.QH.GT.0.) QH=0.
00020 CONTINUE
      TGN=0.
      TGL=0.
      CLAM1=COS(RLAM1)
      SLAM1=SIN(RLAM1)
      FGZ1=FGZ*CLAM1-FGX*SLAM1
      FGX1=FGX*CLAM1+FGZ*SLAM1
      FGZ=FGZ1*CTHT+FGX1*STHT
      FGX=FGX1*CTHT-FGZ1*STHT
      IF(IMODE.EQ.-1)  ICRASH=0
      IF(DZNG.LE.-2..AND.DZMG.LT.-2..AND.IMODE.EQ.1) ICRASH=1
      IF(FGZ.GT.0.) FGZ=0.
      RETURN
      END
      SUBROUTINE ENGINE
     COMMON/GAIN1/GA(100)
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
      COMMON/CHF/B(200)/ICHF/IB(50)
      DIMENSION HDECK(5)
      DIMENSION HIG(6)
      EQUIVALENCE(HUD,B(3)),(THTN,B(17))
      EQUIVALENCE(PLONSTK,B(7))
     EQUIVALENCE(ICOUNTM,IB(47)),(ICOUNTN,IB(48))
      EQUIVALENCE(IMODE,IA(1)),(VJR,B(122)),
    1 (DZCG,A(271)),(CHORD,A(182)),(SPAN,A(181)),
    2(AREA,A(180)),(QBAR,A(178)),(RMACH,A(71)),(VRW,A(70))
      EQUIVALENCE(STAR,B(1)),(STAH1,B(5)),(HUD2,B(188))
      EQUIVALENCE(AIL,B(2))
      EQUIVALENCE(GA(89),RPMMAX)
     EQUIVALENCE(SUTEMPR,A(371)),(RN1,B(19)),(RHO,A(183)),(THTNC,B(16)
    1),(TFPJ,B(44)),(B(45),TRPJ),(B(46),TYAWJ),(TRRJ,B(47)),(TLRJ,B(48
    2 )),(TFPJ1,B(49)),(TRPJ1,B(50)),(TRJ1,B(51)),(TYAWJ1,B(52)),
    3(THROT,B(14)),(ISWWAT,IB(7)),(YSAS,B(12)),(WWDOT1,B(53)),(A(290),
    4 WWAT),(WFUEL,A(249)),(IFUEL,IB(8)),(TAUENG,B(56)),(HDRAG,B(57)),
    5 (B(58),TFN),(B(59),THN),(B(60),VJRCS),(THTN,B(17)),(UGE,B(61)),
    6(ALT1,B(62)),(DXCG,A(198)),(XFN,A(191)),
    7 (ZFN,A(264)),(XRN,A(192)),(ZRN,A(265)),(ZRDRAG,A(263)),
    8  (XHDRAG,A(190)),(XFPJ,A(193)),(XRPJ,A(194)),(XRRJ,A(196)),
    9 (YRRJ,A(222)),(ZYAWJ,A(288)),(XYAWJ,A(195))
      EQUIVALENCE(HIG(1),A(291))
     EQUIVALENCE(ALTGE2,GA(94)),(GEF1,B(63)),(GEF2,B(64)),(B(65),ALTGE1)
    1  )
     EQUIVALENCE(RMACH,A(71))
```

```
      EQUIVALENCE(FCR,B(66)),(TOTHM,B(20)),(AFPJ,B(68)),(ARPJ,B(69))
      EQUIVALENCE(APRJ,B(70)),(ALRJ,B(67))
      EQUIVALENCE           (AYAWJ,B(71)),(CSPLAYF,B(72)),
     1(CSPLAYR,B(73)),(CRJ,B(74)),(CFPJA,B(75)),(CRPJA,B(76)),
     2(SFPJA,B(77)),(SRPJA,B(78)),(SRJ,B(79)),(VJR,B(122))
      EQUIVALENCE(DT,A(168))
      EQUIVALENCE(FEX,A(139)),(FEY,A(140)),(FEZ,A(141)),(TEM,A(159)),
     1(TEL,A(158)),(TEN,A(160))      ,(SALPH,  A(31)),(CALPH,A(32)),
     2  (SBETA,A(33)),(CBETA,A(34))
      EQUIVALENCE(IMODE,IA(1)),(THTNIC,B(18)),(WFUELIC,A(247)),
     1 (WWATIC,A(248))
      EQUIVALENCE(CG,A(256))
      EQUIVALENCE(DXCG0,A(254))
       EQUIVALENCE(THROTIC,B(15))
      EQUIVALENCE(HOFCK(1),A(272))
      EQUIVALENCE(ALT,A(83))
      EQUIVALENCE(B(20),TOTHM)
     EQUIVALENCE(GA(90),THCOR)
     EQUIVALENCE(IMOV,IB(1))
     DATA THCOR/.9933/
     DATA ICOUNTM/140/,ICOUNTN/200/
     DATA GEF1/0./,GEF2/0./
      DATA CSPLAYF/.996/,CSPLAYR/.9778/,CFPJA/.991/,CRPJA/.9902/
      DATA CRJ/1./,SFPJA/-.133/,SRPJA/.1396/,SRJ/0./
     DATA ICOUNTM/140/,ICOUNTN/400/
     DATA RPMMAX/105./
      DATA CG/10.7/
      DATA DXCG0/28.5/
      DATA FCR/.06/
      DATA THTNRAT/150./
      DATA RHOZ/.002377/
     DATA ALTGE1/10./,ALTGE2/20./
      IF(IMODE)81,81,79
00081 CONTINUE
     DDGE1=(GEF2-GEF1)/ALTGE1
     DDGE2=-GEF2/(ALTGE2-ALTGE1)
     ICOUNTE=0
     KCOUNT=0
     NCOUNT=0
      THTN=THTNIC
      IF(SOTEMPR.EQ.0.)SOTEMPR=1.
      THROT=THROTIC
      THTNC=THTNIC
      WWAT=WWATIC
      WFUEL=WFUELIC
      RN1=THROT
      RN1P=RN1
00079 CONTINUE
      DXCG=DXCG0+CHORD*(CG-10.7)/100.
      AFPJ=TAB1(PLONSTK,0.,0.,4,1)
      ARPJ=TAB1(STAB,0.,0.,5,1)
      APRJ=TAB1(AIL,0.,0.,6,1)
      ALRJ=TAB1(-AIL,0.,0.,6,1)
      AYAWJ=TAB1(RUD2,0.,0.,7,1)
      RN2=RN1/SOTEMPR
```

```
      RRHO=RHO/RHOZ
      ATOT=AFPJ+ARPJ+ARRJ+ALRJ+AYAWJ
      PRODUCT=TAB1(ATOT,0.,0.,8,1)
      RN3=RN2/100.
      NSW1=RN3+1.
      GO TO(10,20),NSW1
00010 VJRCS=23.43*RN2
        RMUMAX=23.*RN3*RKNRCS
      GO TO 30
00020 VJRCS=RN2*(-143.24+RN2*(3.3455-.016748*RN2))
      RMUMAX=23.*RKNRCS
00030 CONTINUE
C   NOZZLE ANGLE RESPONSE
      THTNE=THTNC-THTN
      DTHTN=THTNRAT*DT
      IF(ABS(THTNE)-DTHTN)40,40,50
00040 THTN=THTNC
      GO TO 60
00050 THTN=THTN+DTHTN*SIGN(1.,THTNE)
00060 CONTINUE
      IF(THTN-98.5) 90,100,100
00100 THTN=98.5
      GO TO 120
00090 IF(THTN)110,110,120
00110 THTN=0.
00120 CONTINUE
     IF(THTN.LT.20.) GO TO 121
     GO TO 122
00121 SPLAYR=7.5-THTN*.025
     GO TO 125
00122 SPLAYR=8.91245+ THTN*(-.012496+.0008437*THTN)
00125  CSPLAYR=COS(SPLAYR/57.3)
C  REACTION CONTROL EFFECTIVENESS
      THTNR=THTN/20.
      IF(THTN)131,131,129
00129 IF(THTN-20.)130,140,140
00131 RKNRCS=0.
      GO TO 150
00130 RKNRCS=THTNR*(-.100762+THTNR*(3.292+THTNR*(-2.1689-.0223*THTNR)))
      GO TO 150
00140 RKNRCS=1.
00150 CONTINUE
C    RCS THRUST CALCULATION
C    RN4 =DUCT PRESSURE AT NPCT/DUCT PRESSURE AT 100 PCT
      IF(RN3-.5)151,151,152
00151 RN4=.6838*RN3
      GO TO 153
00152 RN4=-.3162+1.3162*RN3
00153 CONTINUE
      FACRCS=RKNRCS*RRHO*RN4*PRODUCT
      RK1FPJ=TAB1(AFPJ,0.,0.,9,1)
      RK1RPJ=TAB1(ARPJ,0.,0.,10,1)
      TFPJ=TFPJ1*FACRCS*RK1FPJ
      TRPJ=TRPJ1*FACRCS*RK1RPJ
      RK1YAWJ=TAB1(AYAWJ,0.,0.,11,1)
```

```
      IF(RUU2)160,160,170
00160 TYAWJ=-TYAWJ1*FACRCS*RK1YAWJ
      GO TO 180
00170 TYAWJ=TYAWJ1*FACRCS*RK1YAWJ
00180 CONTINUE
      IF(AIL)200,210,220
00200 RK1RRJ=TAB1(ARRJ,0.,0.,12,1)
      RK1LRJ=TAB1(ALRJ,0.,0.,13,1)
      TRRJ=TRJ1*FACRCS*RK1RRJ
      TLRJ=-TRJ1*FACRCS*RK1LRJ
      GO TO 230
00210 TRRJ=0.
      TLRJ=0.
      GO TO 230 .
00220 RK1RRJ=TAB1(ARRJ,0.,0.,13,1)
      RK1LRJ=TAB1(ALRJ,0.,0.,12,1)
      TRRJ=-TRJ1*FACRCS*RK1RRJ
      TLRJ=TRJ1*FACRCS*RK1LRJ
00230 CONTINUE
      RMRCS=32.2*(TFRJ+TRPJ+ABS(TYAWJ)+ABS(TRRJ)+ABS(TLRJ))/
     1 (VJRCS+1.)
  6   IF(RMRCS.EQ.0.) GO TO 232
      IF(RMRCS.GT.RMDMAX) GO TO 231
      GO TO 232
00231 CONTINUE
      RMR=RMDMAX/RMRCS
      TFRJ=TFRJ*RMR
      TRPJ=TRPJ*RMR
      TRRJ=TRRJ*RMR
      TLRJ=TLRJ*RMR
      TYAWJ=TYAWJ*RMR
      RMRCS=RMDMAX
00232 CONTINUE
C  ENGINE RPM COMMAND
      RNC=THRUT
      IF(RNC.GT.104..AND.MCOUNT.EQ.1)RNC=104.
      IF(RNC.GT.102..AND.NCOUNT.EQ.1)RNC=102.
C  WATER INJECTION CONTROL
      IF(ISWWAT.EQ.0.)  WWDOT=0.
      IF(ISWWAT.EQ.1..AND.RN1.GT.92.)WWDOT=WWDOT1
      IF(WWAT.LE.0.)WWDOT=0.
      IF(RN1.LT.92.)WWDOT=0.
      WWAT=WWAT-WWDOT*DT
      IF(WWAT.LE.0.)WWAT=0.
C  THRUST COMMAND LIMITER
      IF(RNC.GT.RNMMAX) RNC=RNMMAX
      IF(RNC.GT.102..AND.WWDOT.EQ.0.)RNC=102.
      IF(RNC.GT.102..AND.THTN.LT.10.) RNC=102.
      WFUEL=WFUEL-FCR*RN1*DT
      IFUFL=0
      IF(WFUEL)260,260,270
00270 RNC1=0.
      GO TO 280
00260 RNC1=RNC1+50.*DT
      IFUFL=1
```

```
00280  RNC=RNC-RNC1
       IF(RNC.LE.0.)RNC=0.
       IF(WFUEL.LT.0.)WFUEL=0.
      IF(IMODE)291,291,292
 291  DTAU=.05
      GO TO 293
 292  CONTINUE
       TAUENG=TAB1(RN1,0.,0.,14.1)
      DTAU=DT/TAUENG
 293  CONTINUE
      EXPENG=EXP(-DTAU)
       RN1=RN1+(1.-EXPENG)*(RNC-RN1)
      IF(IHOV.EQ.1)GO TO 281
       IF(RN1.LT.RN1P) GO TO 281
      IF(RN1.LT.70.) DRPM=12.5*DT
      IF(RN1.GT.70..AND.RN1.LT.90.) DRPM=24.*DT
      IF(RN1.GT.90.) DRPM=DT*(111.-RN1)
      RNS=RN1P+DRPM
      IF(RN1.GT.RNS) RN1=RN1P+DRPM
00281  CONTINUE
      RN1P=RN1
C     ENGINE RPM COUNTER
      IF(RN1.GT.102..AND.RN1.LT.104) ICOUNTE=ICOUNTE+1
      IF(RN1.GE.104.) ICOUNTE=ICOUNTE+2
      IF(RN1.LE.100.) ICOUNTE=ICOUNTE-1
      IF(ICOUNTE.LE.0) ICOUNTE=0
      IF(ICOUNTE.LE.0) MCOUNT=0
      IF(ICOUNTE.LE.0) NCOUNT=0
      IF(ICOUNTE.GE.ICOUNTM) MCOUNT=1
      IF(ICOUNTE.GT.ICOUNTN)NCOUNT=1
C  NOZZLE THRUST
      SQRHOR=SQRT(RRHO)
      DGTHRST=.5*TAB1(RN1,RMACH,0.,15,1)*RRHO**3
       TIFN=TAB1(RN1,RMRCS,0.,16,1)
      RDRAG=RRHO*TAB1(RN1,RMACH,0.,18,1)
      TFN=SQRHOR*THCOR*TIFN+DGTHRST
       TIRN=TAB1(RN1,RMRCS,0.,17,1)
      TRN=SQRHOR*THCOR*TIRN+DGTHRST
C    ENGINE AND RCS FORCE AND MOMENT CALCULATION
       SNTHTN=SIN(THTN/57.3+.026178)
       CSTHTN=COS(THTN/57.3+.026178)
C    GROUND EFFECT FACTOR
       ALT1=HIG(2)-HDECK(3)
      IF(ALT1.LT.0.)ALT1=0.
      IF(ALT1-ALTGE1)290,295,295
 290  DGE=GE1+DDGE1*ALT1
      GO TO 320
 295  IF(ALT1-ALTGE2)300,310,310
 300  DGE=GE2+DDGE2*(ALT1-ALTGE1)
      GO TO 320
 310  DGE=0.
 320  CONTINUE
      FEZ=SNTHTN*(-TFN*CSPLAYF-TRN*CSPLAYR)-CFPJA*TFPJ-CHPJA*TRPJ
    1  -CHJ*(THRJ+TLRJ)-RDRAG*SALPH*CHETA
      FEZ=FEZ*(1.+DGE)
```

```
      FEX=CSTHTN*(CSPLAYF*TFN+CSPLAYR*TRN)+SFPJA*TFPJ+SRPJA*TRPJ
     1 -RDRAG*CALPH*CBETA+SRJ*(TRRJ+TLRJ)
      FEY=TYAWJ-RDRAG*SBETA
      TFM=CSPLAYF*TFN*(SNTHTN*(DXCG-XFN)+CSTHTN*(DZCG-ZFN))
     1 +CSPLAYR*TRN*(SNTHTN*(DXCG-XRN)+CSTHTN*(DZCG-ZRN))
     2 -RDRAG*(CALPH*(DZCG-ZRDRAG)-SALPH*(DXCG-XRDRAG))*CBETA
     3 +CFPJA*TFPJ*(DXCG-XFPJ)+CRPJA*TRPJ*(DXCG-XRPJ)
     4 +(TRRJ+TLRJ)*(DXCG-XRRJ)
      TFL=(TLRJ-TRRJ)*YRRJ-TYAWJ*(DZCG-ZYAWJ)  +FEY*(ZRDRAG-DZCG)
      TFN=TYAWJ*(DXCG-XYAWJ)-RDRAG*SBETA*CALPH*(DXCG-XRDRAG)
      TOTRM=TRN+TFN
      RETURN
      END
```

****

```
      SUBROUTINE SHIP
      COMMON/CBF/B(200)/ICBF/IB(50)
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
      COMMON/RK1/TITLE(8),HEAD,SPEED,NSPEC,SIGWH(3),TMODAL(3),
     1 PI,X2PI,HD,NW,DW,NCHN,NREC,KPTS,DELT,SH,TRUN,NPTS,CTITL(19)
      DIMENSION CTJ(13,4),XIJ(13,4),THIST(19,3,30),XLAST(13,3)
      DIMENSION DID(48)
      EQUIVALENCE(TITLE,DID)
      EQUIVALENCE(TIME,A(303)),(RTIME,A(304))
      EQUIVALENCE(DT,A(169))
      EQUIVALENCE(PHIS,B(82)),(THETS,B(83)),(PSIS,B(84)),(THETSD,B(85))
      EQUIVALENCE(PHISD,B(86)),(PSISD,B(87)),(HEAVE,B(88)),(HEAVED,
     1B(89)),(SURGE,B(90)),(SURGED,B(91)),(SWAY,B(92)),(SWAYD,B(93))
      EQUIVALENCE(HMAG,B(104)),(PHIHV,B(106)),(OMEGH,B(107)),
     1(B(105),PHIROL),(OMEGR,B(108)),(ROLLM,B(109)),(PITCHM,B(110))
      EQUIVALENCE(OMEGP,B(111)),(PHIP,B(112)),(YAWM,B(113))
      EQUIVALENCE(OMEGY,B(114)),(PHIY,B(115)),(SURGM,B(116)),(OMEGSRG,
     1B(117)),(PHISRG,B(118)),(SWAYM,B(119)),(OMEGSWY,B(120)),
     2(PHISWY,B(121))
      EQUIVALENCE(COURSE,A(110)),(IMODE,IA(1)),(B(192),WAVHT)
      EQUIVALENCE(ITAPE,IA(24)),(ISTART,IA(25)),(IFILE,IA(26)),
     1   (IA(27),K1),(IFILMX,IA(28)),(JSTART,IA(29))
      DATA SWAYM/3.38/,OMEGSWY/.769/,PHISWY/3.14159/
      DATA HMAG/5.3/,OMEGH/.788/,PHIHV/4.71236/
      DATA OMEGP/.836/
      DATA ROLLM/6.549/,OMEGR/.706/,PHIROL/0./,PITCHM/3.059/
      DATA PHIP/1.57079/,YAWM/.6898/,OMEGY/.753/,PHIY/1.57079/
      DATA SURGM/1.36/,OMEGSRG/.8/,PHISRG/1.57079/
      DATA RED/57.3/
      DATA IFILE/1/
      DATA ITAPE/1/
      DATA IFILMX/6/,ISET/0/,ISTART/1/,JSTART/1/,K1/1/
      IF(IMODEP-IMODE)400,401,400
  400 ISET=0
  401 IF(ITAPE)500,500,501
  500 CONTINUE
      PSISD=YAWM*OMEGY*COS(OMEGY*TIME+PHIY)
```

```
00280  RNC=RNC-RNCI
       IF(RNC.LE.0.)RNC=0.
       IF(WFUEL.LT.0.)WFUEL=0.
       IF(IMODE)291,291,292
  291  DTAU=.05
       GO TO 293
  292  CONTINUE
       TAUENG=TAB1(RN1,0.,0.,14,1)
       DTAU=DT/TAUENG
  293  CONTINUE
       EXPENG=EXP(-DTAU)
       RN1=RN1+(1.-EXPENG)*(RNC-RN1)
       IF(IHOV.EQ.1)GO TO 281
       IF(RN1.LT.RN1P) GO TO 281
       IF(RN1.LT.70.) DRPM=12.5*DT
       IF(RN1.GT.70..AND.RN1.LT.90.) DRPM=24.*DT
       IF(RN1.GT.90.) DRPM=DT*(111.-RN1)
       RNS=RN1P+DRPM
       IF(RN1.GT.RNS) RN1=RN1P+DRPM
00281  CONTINUE
       RN1P=RN1
C      ENGINE RPM COUNTER
       IF(RN1.GT.102..AND.RN1.LT.104) ICOUNTE=ICOUNTE+1
       IF(RN1.GE.104.) ICOUNTE=ICOUNTE+2
       IF(RN1.LE.100.) ICOUNTE=ICOUNTE-1
       IF(ICOUNTE.LE.0) ICOUNTE=0
       IF(ICOUNTE.LE.0) MCOUNT=0
       IF(ICOUNTE.LE.0) NCOUNT=0
       IF(ICOUNTE.GE.ICOUNTM) MCOUNT=1
       IF(ICOUNTE.GT.ICOUNTN)NCOUNT=1
C  NOZZLE THRUST
       SQRHOR=SQRT(RRHO)
       DGTHRST=.5*TAB1(RN1,RMACH,0.,15,1)*RRHO**3
       TIFN=TAB1(RN1,RMRCS,0.,16,1)
       RDRAG=RRHO*TAB1(RN1,RMACH,0.,18,1)
       TFN=SQRHOR*THCOR*TIFN+DGTHRST
       TIRN=TAB1(RN1,RMRCS,0.,17,1)
       TRN=SQRHOR*THCOR*TIRN+DGTHRST
C    ENGINE AND RCS FORCE AND MOMENT CALCULATION
       SNTHTN=SIN(THTN/57.3+.026178)
       CSTHTN=COS(THTN/57.3+.026175)
C    GROUND EFFECT FACTOR
       ALT1=HIG(2)-HDECK(3)
       IF(ALT1.LT.0.)ALT1=0.
       IF(ALT1-ALTGE1)290,295,295
  290  DGF=GEF1+DDGE1*ALT1
       GO TO 320
  295  IF(ALT1-ALTGE2)300,310,310
  300  DGF=GEF2+DDGE2*(ALT1-ALTGE1)
       GO TO 320
  310  DGF=0.
  320  CONTINUE
       FEZ=SNTHTN*(-TFN*CSPLAYF-TRN*CSPLAYR)-CFPJA*TFPJ-CRPJA*TRPJ
     1 -CHJ*(THRJ+TLPJ)-RDRAG*SALPH*CHETA
       FEZ=FEZ*(1.+DGF)
```

```
      FEX=CSTHTN*(CSPLAYF*TFN+CSPLAYR*TRN)+SFPJA*TFPJ+SRPJA*TRPJ
     1  -RDRAG*CALPH*CBETA+SRJ*(TRRJ+TLRJ)
      FEY=TYAWJ-RDRAG*SBETA
      TFM=CSPLAYF*TFN*(SNTHTN*(DXCG-XFN)+CSTHTN*(DZCG-ZFN))
     1 +CSPLAYR*TRN*(SNTHTN*(DXCG-XRN)+CSTHTN*(DZCG-ZRN))
     2 -RDRAG*(CALPH*(DZCG-ZRDRAG)-SALPH*(DXCG-XRDRAG))*CBETA
     3 +CFPJA*TFPJ*(DXCG-XFPJ)+CRPJA*TRPJ*(DXCG-XRPJ)
     4 +(TRRJ+TLRJ)*(DXCG-XRRJ)
      TFL=(TLRJ-TRRJ)*YRRJ-TYAWJ*(DZCG-ZYAWJ)  +FEY*(ZRDRAG-DZCG)
      TFN=TYAWJ*(DXCG-XYAWJ)-RDRAG*SBETA*CALPH*(DXCG-XRDRAG)
      TOTRM=TRN+TFN
      RETURN
     END
```

```
****************************************************************************************
```

```
      SUBROUTINE SHIP
      COMMON/CRF/B(200)/ICBF/IB(50)
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
      COMMON/RK1/TITLE(8),HEAD,SPEED,NSPEC,SIGWH(3),TMODAL(3),
     1 PI,XCPI,HD,NW,DW,NCHN,NREC,KPTS,DELT,SH,TRUN,NPTS,CTITL(19)
      DIMENSION CIJ(13,4),XIJ(13,4),THIST(19,3,30),XLAST(13,3)
      DIMENSION DID(48)
      EQUIVALENCE(TITLE,DID)
      EQUIVALENCE(TIME,A(303)),(RTIME,A(304))
      EQUIVALENCE(DT,A(169))
      EQUIVALENCE(PHIS,B(82)),(THETS,B(83)),(PSIS,B(84)),(THETSD,B(85))
      EQUIVALENCE(PHISD,B(86)),(PSISD,B(87)),(HEAVE,B(88)),(HEAVED,
     1B(89)),(SURGE,B(90)),(SURGED,B(91)),(SWAY,B(92)),(SWAYD,B(93))
      EQUIVALENCE(HMAG,B(104)),(PHIHV,B(106)),(OMEGH,B(107)),
     1(B(105),PHIROL),(OMEGR,B(108)),(ROLLM,B(109)),(PITCHM,B(110))
      EQUIVALENCE(OMEGP,B(111)),(PHIP,B(112)),(YAWM,B(113))
      EQUIVALENCE(OMEGY,B(114)),(PHIY,B(115)),(SURGM,B(116)),(OMEGSRG,
     1B(117)),(PHISRG,B(118)),(SWAYM,B(119)),(OMEGSWY,B(120)),
     2(PHISWY,B(121))
      EQUIVALENCE(COURSE,A(110)),(IMODE,IA(1)),(B(192),WAVHT)
      EQUIVALENCE(ITAPE,IA(24)),(ISTART,IA(25)),(IFILE,IA(26)),
     1   (IA(27),K1),(IFILMX,IA(28)),(JSTART,IA(29))
      DATA SWAYM/3.38/,OMEGSWY/.769/,PHISWY/3.14159/
      DATA HMAG/5.3/,OMEGH/.786/,PHIHV/4.71238/
      DATA OMEGP/.836/
      DATA ROLLM/6.849/,OMEGR/.706/,PHIROL/0./,PITCHM/3.059/
      DATA PHIP/1.57079/,YAWM/.6898/,OMEGY/.753/,PHIY/1.57079/
      DATA SURGM/1.38/,OMEGSRG/.8/,PHISRG/1.57079/
      DATA RCD/57.3/
      DATA IFILE/1/
      DATA ITAPE/1/
      DATA IFILMX/6/,ISET/0/,ISTART/1/,JSTART/1/,K1/1/
      IF(IMODEP-IMODE)400,401,400
400   ISET=0
401   IF(ITAPE)500,500,501
500   CONTINUE
      PSISD=YAWM*OMEGY*COS(OMEGY*TIME+PHIY)
```

```
      PHISD=ROLLM*OMEGR*COS(TIME*OMEGR+PHIROL)
      THETS=PITCHM*SIN(OMEGP*TIME+PHIP)
      THETSD=PITCHM*OMEGP*COS(TIME+PHIP)
      PSIS=YAWM*SIN(OMEGY*TIME+PHIY)
      PSIS=PSIS+COURSE
      PHIS=ROLLM*SIN(OMEGR*TIME+PHIROL)
      SURGE=SURGM*SIN(OMEGSRG*TIME+PHISRG)
      SURGEU=SURGM*OMEGSRG*COS(OMEGSRG*TIME+PHISRG)
      SWAY=SWAYM*SIN(OMEGSWY*TIME+PHISWY)
      SWAYD=SWAYM*OMEGSWY*COS(OMEGSWY*TIME+PHISWY)
      HEAVE=HMAG*SIN(OMEGH*TIME+PHIHV)
      HEAVEU=HMAG*OMEGH*COS(OMEGH*TIME+PHIHV)
      GO TO 1000
501   IF(IMODE)502,502,503
502   IF(ISET)504,504,1000
504   ISET=1
      T1=0.
      T2=0.
      IF(IFILE.GT.IFILMX)IFILE=IFILMX
      DO 600 I=1,IFILE
      JKFILE=0
1     JKFILE=JKFILE+1
      READ(10) DID
      IF(EOF(10))1,2
2     CONTINUE
      IF(I-IFILE) 520,510,520
510   WRITE(6,3)JKFILE
3     FORMAT(10X,*JKFILE*,I10)
      WRITE(6,1005)TITLE
1005 FORMAT(*1*,19X,8A10///)
      WRITE(6,1006) HEAD,SPEED,SR,TRUN,SIGWH,TMODAL,K1
1006 FORMAT(30X,*HEADING =*,F6.0,* DEG*,18X,*SPEED *,F6.2,* KNOTS*  /
     130X,* SAMPLE RATE*,F6.2,* SAM/SEC*,10X,* RUN TIME = *,F8.2,* SEC*
     1///,18X,3(8X,* SIGNIF. WF= *,F6.2,* FEET*,2X,)/,18X,3(8X,*MODAL
     2 PER= *,F5.1,* SEC *,2X)//,6X,* K1 = *,I5)
      WRITE(6,1007)PI,X2PI,RD,NW,DW,NCHN,NREC,KPTS,DELT,SR,TRUN,NPTS
1007 FORMAT(10X,* PI= *,F10.5,* X2PI=*,F10.5,* RD= *,F10.5,/,
     1 10X,* NW = *,I10,* DW= *,F10.5,* NCHN= *,I10,* NREC= *,I10,/,
     2 10X,* KPTS= *,I10,* DT= *,F10.5,* SR= *,F10.5,* TRUN= *,F10.5,
     3 * NPTS= *,I10/)
      IF(JSTART.LT.1) JSTART=1
      KMAX=KPTS-3
      IF(JSTART.GT.KMAX)JSTART=KMAX
      GO TO 530
520   DO 100 NN=1,NREC
      READ(10) THIST
100   CONTINUE
      GO TO 600
530   DO 200 KK=1,ISTART
      READ(10) THIST
200   CONTINUE
600   CONTINUE
      H1=1./(2.*DELT)
      C1=-.5/(DELT*DELT)
      G1=.5/(DELT*DELT*DELT)
```

```
      DO 201 K=1,3
      DO 201 L=1,13
      XLAST(L,K)=THIST(L,K1,K-3+KPTS)
201   CONTINUE
      J=JSTART+1
      WAVHT=THIST(1,K1,J)
      SURGE=THIST(2,K1,J)
      SWAY=THIST(3,K1,J)
      HEAVE=THIST(4,K1,J)
      PHIS=THIST(5,K1,J)
      THETS=THIST(6,K1,J)
      PSIS=THIST(7,K1,J)+COURSE
      SURGED=THIST(8,K1,J)
      SWAYD=THIST(9,K1,J)
      HEAVED=THIST(10,K1,J)
      PHISD=THIST(11,K1,J)
      THETSD=THIST(12,K1,J)
      PSISD=THIST(13,K1,J)
      GO TO 1000
503   CONTINUE
      IF(TIME-T1) 801,800,800
800   T1=T1+DELT
      IF(J-KPTS+2)804,804,806
803   CONTINUE
      IF(J.EQ.(KPTS+2))J=2
804   DO 805 K=1,4
      DO 805 L=1,13
      XIJ(L,K)=THIST(L,K1,J-2+K)
805   CONTINUE
      IF(J.EQ.(KPTS-2))READ(10) THIST
      J=J+1
      GO TO 830
806   IF(J-KPTS+1) 808,808,810
808   DO 809 L=1,13
      XIJ(L,1)=XLAST(L,1)
      XIJ(L,2)=XLAST(L,2)
      XIJ(L,3)=XLAST(L,3)
      XIJ(L,4)=THIST(L,K1,1)
809   CONTINUE
      J=J+1
      GO TO 830
810   IF(J-KPTS)811,811,815
811   DO 812 L=1,13
      XIJ(L,1)=XLAST(L,2)
      XIJ(L,2)=XLAST(L,3)
      XIJ(L,3)=THIST(L,K1,1)
812   XIJ(L,4)=THIST(L,K1,2)
      J=J+1
      GO TO 830
815   IF(J-KPTS-1)816,816,803
816   DO 817 L=1,13
      XIJ(L,1)=XLAST(L,3)
      XIJ(L,2)=THIST(L,K1,1)
      XIJ(L,3)=THIST(L,K1,2)
      XIJ(L,4)=THIST(L,K1,3)
```

```
817   CONTINUE
      J=J+1
830   CONTINUE
      IF(J.EQ.3) GO TO 820
      GO TO 823
820   DO 822 K=1,3
      DO 822 L=1,13
      XLAST(L,K)=THIST(L,K1,K-3+KPTS)
822   CONTINUE
823   CONTINUE
      DO 840 L=1,13
      CIJ(L,1)=XIJ(L,2)
      CIJ(L,2)=H1*(XIJ(L,3)-XIJ(L,1))
      CIJ(L,3)=C1*(XIJ(L,4)-4.*XIJ(L,3)+5.*XIJ(L,2)-2.*XIJ(L,1))
      CIJ(L,4)=D1*(XIJ(L,4)-3.*XIJ(L,3)+3.*XIJ(L,2)-XIJ(L,1))
840   CONTINUE
801   CONTINUE
      IF((TIME-T2).GE.DELT)T2=T2+DELT
      DTI=TIME-T2
      DTI2=DTI*DTI
      DTI3=DTI2*DTI
      WAVHT=CIJ(1,1)+CIJ(1,2)*DTI+CIJ(1,3)*DTI2+CIJ(1,4)*DTI3
      SURGE=CIJ(2,1)+CIJ(2,2)*DTI+CIJ(2,3)*DTI2+CIJ(2,4)*DTI3
      SWAY=CIJ(3,1)+CIJ(3,2)*DTI+CIJ(3,3)*DTI2+CIJ(3,4)*DTI3
      HEAVE=CIJ(4,1)+CIJ(4,2)*DTI+CIJ(4,3)*DTI2+CIJ(4,4)*DTI3
      PHIS=CIJ(5,1)+CIJ(5,2)*DTI+CIJ(5,3)*DTI2+CIJ(5,4)*DTI3
      THETS=CIJ(6,1)+CIJ(6,2)*DTI+CIJ(6,3)*DTI2+CIJ(6,4)*DTI3
      PSIS=CIJ(7,1)+CIJ(7,2)*DTI+CIJ(7,3)*DTI2+CIJ(7,4)*DTI3+COURSE
      SURGED=CIJ(8,1)+CIJ(8,2)*DTI+CIJ(8,3)*DTI2+CIJ(8,4)*DTI3
      SWAYD=CIJ(9,1)+CIJ(9,2)*DTI+CIJ(9,3)*DTI2+CIJ(9,4)*DTI3
      HEAVED=CIJ(10,1)+CIJ(10,2)*DTI+CIJ(10,3)*DTI2+CIJ(10,4)*DTI3
      PHISD=CIJ(11,1)+CIJ(11,2)*DTI+CIJ(11,3)*DTI2+CIJ(11,4)*DTI3
      THETSD=CIJ(12,1)+CIJ(12,2)*DTI+CIJ(12,3)*DTI2+CIJ(12,4)*DTI3
      PSISD=CIJ(13,1)+CIJ(13,2)*DTI+CIJ(13,3)*DTI2+CIJ(13,4)*DTI3
      IF(TIME.GE.RTIME)REWIND 10
1000  CONTINUE
      IMODEK=IMODE
      RETURN
      END
```

****

```
      SUBROUTINE DECK
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
      COMMON/CHF/B(200)/ICHF/IB(50)
      DIMENSION ITOUCH(5)
      DIMENSION VDKN(7),VDKE(7)
      DIMENSION HDTDKF(7)
      DIMENSION XDTDKD(7),YDTDKD(7),HDTDKD(7),DXPA(15),DYPA(15)
      DIMENSION DZPA(15),HDECK(7),HIG(6),XIG(6),YIG(6),XIGD(6),YIGD(6)
      EQUIVALENCE(HE,B(140)),(XE,B(141)),(YE,B(142))
      EQUIVALENCE(DXCG,A(174)),(DYCG,A(175)),(HCG,A(176)),(A(16),T11),
     1(T21,A(17)),(T31,A(18)),(T12,A(19)),(T22,A(20)),(T32,A(21)),
```

```
     2(T13,A(22)),(T23,A(23)),(T33,A(24)),(DXPA(1),A(186)),
     3(DYPA,A(210)),(DZPA(1),A(257)),(HDECK(1),A(272)),(XIG(1),
     4A(279))
      EQUIVALENCE(YIG(1),A(265)),(HIG(1),A(291)),(VX,A(200)),(VY,A(201))
      EQUIVALENCE(XCGP,A(203)),(YCGP,A(204))
      EQUIVALENCE(ICRASH,IB(9)),(RLAM1,B(80)),(RLAM2,B(81))
      EQUIVALENCE(PHIS,B(82)),(THETS,B(83)),(PSIS,B(84)),(THETSD,B(85))
      EQUIVALENCE(PHISD,B(86)),(PSISD,B(87)),(HEAVE,B(88)),(HEAVED,
     1B(89)),(SURGE,B(90)),(SURGED,B(91)),(SWAY,B(92)),(SWAYD,B(93))
      EQUIVALENCE(X1S,B(94)),(X2S,B(95)),(Y1S,B(96)),(Y2S,B(97))
      EQUIVALENCE(X3S,B(98)),(H3S,B(99)),(H1S,B(100)),(XTD,B(101))
      EQUIVALENCE(YTD,B(102))
      EQUIVALENCE(XTDI,B(189)),(YTDI,B(190))
      EQUIVALENCE(ALT,A(83))
      EQUIVALENCE(IOVDK,IA(15))
      EQUIVALENCE(ILAND,IB(3)),(ITOUCH,IA(10))
      EQUIVALENCE(H1PS,B(103))
      EQUIVALENCE(DNR,A(106)),(DER,A(107))
      EQUIVALENCE(VDKN(1),A(433)),(HDTDKE(1),A(447))
      EQUIVALENCE(HTD,B(31))
      DATA X1S/-200./,X2S/0./,Y1S/-50./,Y2S/50./,X3S/400./,H3S/100./
      DATA H1S/30./,XTD/-100./,YTD/0./,HCGLG/4.6/,H1PS/30./
      DATA ILAND/1/
C     COMPUTE GEAR LOCATIONS IN INERTIAL AXES
      DO 10 J=1,4
      XIG(J)=DNR+DXPA(J)*T11+DYPA(J)*T21+DZPA(J)*T31
      YIG(J)=DER+DXPA(J)*T12+DYPA(J)*T22+DZPA(J)*T32
      HIG(J)=ALT-DXPA(J)*T13-DYPA(J)*T23-DZPA(J)*T33
   10 CONTINUE
      IOVDK=0
      IF(ILAND)100,100,1
C     ILAND=0 MEANS LAND ON NORTH RUNWAY,ILAND=1 MEANS LAND ON SHIP
    1 CONTINUE
      CALL SHIP
C     COMPUTE AIRCRAFT CG LOCATIONS IN DECK AXES
      DATA D2R/.01759/
      PSISR=PSIS*D2R
      CPSIS=COS(PSISR)
      SPSIS=SIN(PSISR)
      XTD=DNR*CPSIS+DER*SPSIS-SURGE
      YTD=DER*CPSIS-DNR*SPSIS-SWAY
      THETSR=THETS*D2R
      CTHTS=COS(THETSR)
      STHTS=SIN(THETSR)
      PHISR=PHIS*D2R
      SPHIS=SIN(PHISR)
      CPHIS=COS(PHISR)
      T11S=CTHTS*CPSIS
      T21S=SPHIS*STHTS*CPSIS-CPHIS*SPSIS
      T31S=CPHIS*STHTS*CPSIS+SPHIS*SPSIS
      T12S=CTHTS*CPSIS
      T22S=SPHIS*STHTS*SPSIS+CPHIS*CPSIS
      T32S=CPHIS*STHTS*SPSIS-SPHIS*CPSIS
      T13S=-STHTS
      T23S=SPHIS*CTHTS
```

```
      T33S=CPHIS*CTHTS
C     COMPUTS GEAR LOCATIONS IN THE THET=0,PHI=0,PSIS AXES
      DO 6 J=1,4
      XIGD(J)=XIG(J)*CPSIS+YIG(J)*SPSIS-SURGE
      YIGD(J)=YIG(J)*CPSIS-XIG(J)*SPSIS-SWAY
 6    CONTINUE
C     COMPUTE DECK HEIGHT BENEATH GEARS AND CG
      HDECK(1)=HEAVE+HIPS*T33S-XID*T13S-YID*T23S+HIS-HIPS
      DO 7 J=1,4
      HDECK(J+1)=HEAVE-XIGD(J)*T13S-YIGD(J)*T23S+HIPS*(T33S-1.)+HIS
 7    CONTINUE
C     COMPUTE DISTANCE FROM CG TO TOUCHDOWN POINT IN NORTH EAST AXES
      XTDI=(XTD+SURGE)*CPSIS-(YTD+SWAY)*SPSIS
      YTDI=(XTD+SURGE)*SPSIS+(YTD+SWAY)*CPSIS
      XF=DNN-XTDI
      YF=DEN-YTDI
      HTD=HEAVE-XTD*T13S-YTD*T23S+HIPS*T33S+HIS-HIPS
      HF=HIG(2)-HTD
      IF(XTD-XIS)100,2,2
 2    IF(XTD-XAS)3,3,100
 3    IF(YTD-YIS)100,100,4
 4    IF(YTD-YAS)5,5,100
 5    CONTINUE
      IOJOK=1
C     COMPUTE X,Y VELOCITIES OF DECK TOUCHDOWN POINTS TO DETERMINE RELATIVE
C     DECK GEAR VELOCITIES
      XDTDKD(1)=-HIPS*THETSD-YID*PSISD
      YDTDKD(1)=XID*PSISD+HIPS*PHISD
      HDTDKD(1)=XID*THETSD-YID*PHISD
      DO 8 J=1,4
      XDTDKD(J+1)=-HIPS*THETSD-YIGD(J)*PSISD
      YDTDKD(J+1)=XIGD(J)*PSISD+HIPS*PHISD
      HDTDKD(J+1)=XIGD(J)*THETSD-YIGD(J)*PHISD
 8    CONTINUE
      SURGEDN=SURGED*CPSIS-SWAYD*SPSIS
      SURGEDE=SURGED*SPSIS+SWAYD*CPSIS
      DO 9 J=1,5
      VDKN(J)=XDTDKD(J)*T11S+YDTDKD(J)*T21S-HDTDKD(J)*T31S+SURGEDN
      VDKE(J)=XDTDKD(J)*T12S+YDTDKD(J)*T22S-HDTDKD(J)*T32S+SURGEDE
      HDTDKE(J)=-XDTDKD(J)*T13S-YDTDKD(J)*T23S+HDTDKD(J)*T33S+HEAVED
 9    CONTINUE
      YWG=DYPA(3)
      XMG=DXPA(1)
      XNG=DXPA(2)
      FLAM1=(HDECK(5)-HDECK(4))/(2.*YWG)
      FLAM2=(HDECK(2)-HDECK(3))/(XNG-XMG)
      GO TO 101
100   DO 11 J=1,5
      HDECK(J)=0.
      VDKN(J)=0.
      VDKE(J)=0.
      HDTDKE(J)=0.
11    CONTINUE
      FLAM1=0.
      FLAM2=0.
```

```
101   CONTINUE
      DO 14 J=1,4
      IF(HIG(J)-HDECK(J+1))12,12,13
12    ITOUCH(J)=1
      GO TO 14
13    ITOUCH(J)=0
14    CONTINUE
      RETURN
      END
```

****

```
      SUBROUTINE PILOT
      COMMON/XFLOAT/A(500)/IFIXED/IA(200)
      COMMON/CHF/B(200)/ICBF/IB(50)
      COMMON/GAIN1/GA(100)
      DIMENSION THETEL(20),QBL(20),XEIL(20),VYL(20),PHIEL(20),NLAG(20),
     1PBL(20),PSIFL(20),RHL(20),ALTEL(20),ALTDEL(20),ALTEIL(20),ALTDDEL(
     1 20),YEIL(20)
      DIMENSION VXL(20)
      DIMENSION HDTUKE(7)
      EQUIVALENCE(IA(1),IMODE)
      EQUIVALENCE(THETH,A(5)),(PSIR,A(6)),(PHIR,A(4))
      EQUIVALENCE(ALFA,A(25)),(BETA,A(26)),(PB,A(37)),(QB,A(38))
      EQUIVALENCE(XCG,A(174)),(YCG,A(175)),
     1  (A(39),RH),(VRW,A(70)),(ALTD,A(80)),(ALT,A(83))
      EQUIVALENCE(IBRAK,IB(29)),(IOVDK,IA(15)),( BLONSTK,B(7))
      EQUIVALENCE(BLNSTKO,B(8)),(BLATSTK,B(9)),(BLTSTKO,B(10)),
     1  (RUDPED,B(11)),(THROT,B(14)),(THROTIC,B(15)),(THTNC,B(16))
      EQUIVALENCE(RPEDO,B(21)),(HE,B(140)),(XE,B(141)),(YE,B(142))
      EQUIVALENCE(ALTCOM,B(143)),(ALTDCOM,B(144)),(YCOM,B(145))
      EQUIVALENCE(XCOM,B(146)),(THETCOM,B(147)),(      PHICOM,B(148))
      EQUIVALENCE(VXCOM,B(149)),(XTDI,B(189)),(YTDI,B(190))
      EQUIVALENCE(CPSI,A(15)),(SPSI,A(14)),(PHIIC,A(230))
      EQUIVALENCE(THETIC,A(231))
      EQUIVALENCE(ITASK,IB(30))
      EQUIVALENCE(VE,A(65)),(VN,A(64)),(CTHETH,A(115)),(STHETR,A(114))
      EQUIVALENCE(ALFAIC,B(42))
     EQUIVALENCE(NLAG(1),IA(30))
      EQUIVALENCE(VX,A(200)),(VY,A(201)),(THTNIC,B(18))
      EQUIVALENCE(YCG,A(175)),(DT,A(168))
     EQUIVALENCE(IHNDOFF,IB(5))
      EQUIVALENCE(IHOLD,IB(17)),(ISET,IB(12))
      EQUIVALENCE(ISWWAT,IB(7))
      EQUIVALENCE(TIME,A(303))
     EQUIVALENCE(HDTOKE(1),A(447))
      EQUIVALENCE(NCHK,IB(22)),(FREQCK,GA(18)),(T0,GA(19)),(T1,GA(20))
      EQUIVALENCE(B(150),PSICOM)
      EQUIVALENCE(T2,GA(21)),(RMAG,GA(22)),(MCHK,IB(23))
      EQUIVALENCE(INDEXT,IA(110))
      EQUIVALENCE(XCG,A(174))
      EQUIVALENCE(VR,A(454))
      EQUIVALENCE(GA(36),VERI1),(GA(37),ALT1),(GA(38),RKIV),(GA(39),
```

```
1  RK2V)
 EQUIVALENCE(GA(41),RKRUD1),(GA(42),RKRUD2)
 EQUIVALENCE(GA(43),ALT2),(GA(44),RKTHT1),(GA(45),RKALPH)
  EQUIVALENCE(GA(46),THET1),(GA(47),ALT3),(GA(48),RKALTD)
 EQUIVALENCE(GA(49),THETCO),(RKTHT2,GA(50)),(RKTHT3,GA(51)),(RKTHT4
1  ,GA(52))
 EQUIVALENCE(RKYE,GA(53)),(RKVE,GA(54)),(RKPSI,GA(55))
 EQUIVALENCE(GA(56),RKPHILT),(RKPLT,GA(57))
 EQUIVALENCE(GA(58),RKTHLN),(GA(59),RKQLN),(GA(60),RKNOZLN)
 EQUIVALENCE(GA(61),VRW1),(GA(62),ALT4),(GA(63),THTNC1),(GA(64),
1 XCG1),(GA(65),VRW2),(GA(66),VRW3),(GA(67),VRW4),(GA(68),ALT5)
 EQUIVALENCE(GA(69),RKTHT5),(RKVY1,GA(70)),(RKVX1,GA(71))
 EQUIVALENCE(RKVY2,GA(72)),(RKVX2,GA(73)),(GA(74),RKYPHI),(GA(75),
1  RKPHIVY)
 EQUIVALENCE(GA(76),RLNTHT2),(GA(77),RLNQB2)
 EQUIVALENCE(GA(78),RLTPHI2),(GA(79),RLTPB2)
 EQUIVALENCE(GA(80),RPR1),(GA(81),RPPSI1)
 EQUIVALENCE(HIC,A(241))
 EQUIVALENCE(GA(82),RTHROT),(GA(83),RTHROT1),(GA(84),RTHROT2)
 EQUIVALENCE(GA(85),ALFTHM1)
 EQUIVALENCE(GA(86),XCG2)
   EQUIVALENCE(GA(87),TPITCH),(GA(88),THET2)
 EQUIVALENCE(PSAS,H(6)),(STAH,H(1))
 EQUIVALENCE(I200F,IA(16))
 EQUIVALENCE(HTD,H(31))
 EQUIVALENCE(IHTRACK,IH(32))
  EQUIVALENCE(HHOV,H(32))
 EQUIVALENCE(VDD,A(90)),(RTHROT3,GA(91))
 EQUIVALENCE(RALTTHR,GA(92)),(RALTDT,GA(93))
 EQUIVALENCE(A(500),PSIW)
 EQUIVALENCE(HIC,     A(241))
 EQUIVALENCE(GA(95),THRESH1),(GA(96),THRESH2),(GA(97),THRESH3),
1 (GA(98),THRESH4),(GA(99),THX),(GA(100),TRY)
 EQUIVALENCE(XE1,A(499)),(YE1,A(498)),(XE2,A(497)),(YE2,A(496))
 EQUIVALENCE(RKTHT6,A(495)),(RKYPMAX,A(494))
 EQUIVALENCE(RLTAF,A(484)),(RLTVF,A(483)),(RLTSC,A(482))
 EQUIVALENCE(TAF,A(493)),(TVF,A(492)),(TSCPSEC,A(491))
 EQUIVALENCE(RLNAF,A(487)),(RLNVF,A(486)),(RLNSC,A(485))
 EQUIVALENCE(PEDAF,A(490)),(PEDVF,A(489)),(PEDSC,A(488))
 DATA RALTTHR/.2/,RALTDT/-.5/
 DATA IHTRACK/0/
  DATA HHOV/10./
 DATA THET1/.1396/
 DATA XCG2/500./
 DATA ALT2/50./
 DATA RKTHT5/.01/,RKTHT6/.02/
 DATA THX/3./,THY/5./,RKYPMAX/-.03/
 DATA RKYPHI/-.01/,RLNTHT2/15./,RLNQB2/10.5/
 DATA DTHROT/.5/
 DATA RLTPB2/-10./,RPR1/5./,RPPSI1/2./
 DATA RLTPHI2/-10./
 DATA ALFTHM1/8./
 DATA TPITCH/15./,THET2/.209   /
  DATA VRW11/10./,ALT1/300./,RK1V/3./,RK2V/1./
 DATA IHHOFF/0/
```

```
      DATA VRR1/270./
      DATA RKRUD1/5./,RKRUD2/-.1/
      DATA ALFTHM1/8./
      DATA RKTHT1/.01/,RKALPH/-.5/
      DATA ALT3/1000./,RKALTD/.25/
      DATA T2P/.1/,T3P/.1/,T4P/.2/,T5P/.2/
      DATA(NLAG(I),I=1,14)/14*1/
      DATA THRESH1/.05/,THRESH2/.05/,THRESH3/.25/,THRESH4/.25/
      DATA THETCO/.02/,RKTHT2/-.005/,RKTHT3/-.01/,RKTHT4/-.0005/
      DATA RKVF/-.002/,RKVF/-.01/,RKPSI/-1./
      DATA RKPHILT/-20./,RKPLT/-12./
      DATA RKTHLN/25./,RKOLN/7.5/,RKNOZLN/.02/
      DATA ALT4/80./,THTNC1/55./,XCG1/350./
      DATA VRR2/200./,VRR3/230./,VRR4/300./,ALT5/600./
      DATA RTHROT1/-2./,RTHROT1/-3./,RTHROT2/0./
      DATA T0/1./,T1/2./,T2/3./,RMAG/1./
      DATA RLNSTK0/.6/
      DATA RTHROT3/0./
      DATA TDEL1/.4/,TDEL2/.4/
      IF(IMODE)1,1,2
    1 RLONSTK=RLNSTK0
      HTDC=HTD
      IBRAK=1
      TTHROT=2.
      ISWAT=1
      ISWATCH=1
      ISWATC=1
      TDEL3=TDEL1+TDEL2
      EXP2LN=EXP(-DT/T2P)
      EXP2LT=EXP(-DT/T3P)
      EXP2P=EXP(-DT/T4P)
      EXP2T=EXP(-DT/T5P)
      PLONST2=0.
      PLAT2=0.
      RUDP2=0.
      THROT2=0.
      THROTO=0.
      IHSW=0
      RUDOLD=0.
      PLTIOLD=0.
      PLNIOLD=0.
      IVZ=0
      IVSW=0
      IVSW=0
      PEDSC1=0.
      PEDAF=0.
      PEDVF3=0.
      PLNAF=0.
      RLNVF3=0.
      PLNSC1=0.
      TAF=0.
      TVF3=0.
      TSC=0.
      PLTAF=0.
      PLTVF3=0.
```

```
      PLTSC1=0.
      IF(ALT-ALT3+10.)7,6,6
6     THETCO=THETR
       VXCOM=VHW
      THETRC1=0.
      ICLIMB=0
      SWITCH=0.
      IHOLD=1
      IALTI=1
      GO TO 8
7     CONTINUE
       IALTI=0
      ICLIMB=1
       IHOLD=0
      THETHC2=0.
       THETHC1=0.
8     CONTINUE
      THETRCP=THETR
      CPSIW=COS(PSIW)
      SPSIW=SIN(PSIW)
      XCOMO=XE*CPSI+YE*SPSI
      IF(ABS(XCOMO).LT.1.)XCOMO=SIGN(1.0,XCOMO)
      YCOMO=YE*CPSI-XE*SPSI
      VXCOMO=VX
      XE2=XCOMO
      YE1=YCOMO
      IF(SIGN(1.,VXCOMO).EQ.SIGN(1.,XCOMO))VXCOMO=-.1*XCOMO
      VDECKP=HDTDKE(1).
       VERI=0.
       SETHHAP=0
       THETCOM=THETIC
      XCOM=XCOMO
       ALFTRM=ALFA
       PHICOM=0.
      YCOM=YCOMO
       VOWP=VHW
       ALTCOM=ALT
         ALTDCOM=0.
       THTHC=THTHIC
       PLATSTK=PLTSTKO
       ALTFI=0.
       THROT=THROTIC
     THRTMIN=THROTIC-5.
     THLIM1=100.-THROTIC
     THLIM2=102.-THROTIC
     THLIM3=THRTMIN-THROTIC
       WIDPEU=WPEDO
       GO TO 100
2      CONTINUE
       GO TO (10,20,30,40,50,60),ITASK
10     CONTINUE
       THROT=200.*TIME+50.
       IF(THROT.GT.100.) THROT=100.
       PLONSTK=PLNSTKO
       PLATSTK=0.
```

```
      IBRAK=1
      IF(TIME.GT.2.)IBRAK=0
      ISWAT=0
      THTNC=0.
      IF(VRW.GT.100.)SETBRAK=1.
      IF(SETBRAK.EQ.1.)IBRAK=1
      IF(SETBRAK.EQ.1.)TRET=1.
      IF(TRET.EQ.1.)THROT=0.
      THWAA=20.
      GO TO 110
20    CONTINUE
      YF=YCG
      CNOZ=COS(THTNC/57.3)
      SNOZ=SIN(THTNC/57.3)
      IBRAK=0
      IF(TIME.LE.2.) IBRAK=1
      IF(VW.GT..25) IBRAK=0
      VER=VXCOM-VRW
      VERI=VERI+VER*DT
      IF(ABS(VERI).GT.VERI1)VERI=SIGN(VERI1,VERI)
      IF(ALT.LT.ALT1) VERI=0.
      THROTOL=THROT
      THROT=THROTIC+(RK1V*VER+RK2V*VERI)*CNOZ+SNOZ*(RALTTHR*(ALTCOM-ALT)
     1 +ALTDF*RALTDT)
      IF(ALT.LT.ALT5.AND.THROT.LT.100.) THROT=100.
      IF(THROT.GT.(THROTOL+DTHROT))THROT=THROTOL+DTHROT
      IF(THROT.LT.(THROTOL-DTHROT))THROT=THROTOL-DTHROT
      IF(VRW.LT.VRW1) THROT=106.
      RUDPED=RKRUD1*RA+RKRUD2*BETA+RPEDO
      IF(IHOLD)31,31,32
31    CONTINUE
C       PITCH COMMAND
      IF(ALFTRM.LT.ALFTRM1)ALFTRM=ALFTRM+.2
      ALTDCOM=ALTD
      IF(ALTDCOM.LT.0.)ALTDCOM=0.
      THETRC3=ALFTRM/57.3+ALTDCOM/VRW
      THETRC2=THETRC2+.3*(THETRC3-THETR)*DT
      THETRC=THETRC2+THETRC3
      ALTCOM=ALT
      ALTDCM1=ALTDCOM
      SWITCH=1
      IF(THETRC.LT.THET1)THETRC=THET1
      IF(THETRC.GT.THET2.AND.(THTNC.GT.10..OR.XCG.LT.XCG1))THETRC=THET2
      IF(ALT.GT.(ALT3-(ALTD**2)/20.)) ALT5=ALT
      THETRC1=THETRC
      IF(THETRC1.GT..5)THETRC1=.5
      THETCO=THETRC1-ALFA/57.3
      GO TO 33
32    CONTINUE
      SWITCH=SWITCH*.95
      THETRC1=THETRC1*.95
      ALTDCOM=ALTDCM1*(1.-((ALT-ALT5)/(ALT3-ALT5))**2)
      IF(IALT1.EQ.1)ALTDCOM=0.
      ALTCOM=ALT
      IF(ABS(ALT-ALT3).LT.20.) IALT1=1
```

```
      IF(IALTI.EQ.1) ALTCOM=ALT3
       ALTE=ALT-ALTCOM
       ALTDE=ALTD-ALTDCOM
      IF(ABS(ALTE).GT.50.)ALTE=SIGN(50.,ALTE)
      IF(ABS(ALTDE).GT.30.)ALTDE=SIGN(30.,ALTDE)
      IF(ALT.GT.ALT3) IALTI=1
       ALTEI=ALTEI+ALTE*DT
       IF(ABS(ALTEI).GT.200.) ALTEI=SIGN(200.,ALTEI)
      THETRC=THETRC1+(1.-SWITCH)*(THETCO+CNOZ*(RKTHT2*ALTE
     1  +RKTHT3*ALTDE+RKTHT4*ALTEI))
      DTH=THETRC-THETRCP
      IF(ABS(DTH).GT..01)THETRC=THETRCP+SIGN(.01,DTH)
      IF(THETRC.LT.THETRCP.AND.ALFA.LT.1.)THETRC=THETRCP
33     CONTINUE
      IF(THETRC.GT..5)THETRC=.5
      IF(THETRC.LT..1)THETRC=.1
      THETRCP=THETRC
        THETCOM=THETRC*57.3
      PHIC=RKYE*YE+RKVE*(VE*CTHETR-VN*STHETR)+RKPSI*PSIR
       IF(ABS(PHIC).GT..4)PHIC=SIGN(.4,PHIC)
       PHICOM=PHIC*57.3
      RLATSTK=RKPHILT*(PHIR-PHIC)+RKPLT*PB+RLTSTKO
      RLONSTK=RKTHLN*(THETR-THETRC)+RKQLN*QB+RLNSTKO+RKNOZLN*(THTNC-
     1   THTNIC)
C     TRIM FOLLOWER
      RLNSTKO=RLNSTKO+(RLONSTK-RLNSTKO)*.004*(IMODE+1)
       ISWWAT=1
      IF(INOEXT.EQ.1.AND.XCG.GT.XCG1.AND.ALT.LT.ALT4)THTNC=THTNC1
      IF(ALT.GT.ALT4.AND.ALTD.GT.0..AND.VRW.GT.VRW2)THTNC=THTNC-2.5*DT
      IF(ALT.GT.ALT4.AND.ALTD.GT.0..AND.VRW.GT.VRW3)THTNC=THTNC-5.*DT
      IF(ALT.GT.ALT4.AND.ALTD.GT.0..AND.VRW.GT.VRW4) THTNC=THTNC-10.*DT
       IF(THTNC.LT.0.)THTNC=0.
      IF(ALT.GE.ALT5)IHOLD=1
      IF(INNOFF.EQ.1.AND.XCG.LT.XCG2.AND.THETR.LT..26.AND.THETR.GT..07.
     1  AND.ABS(PHIR).LT..2) GO TO 34
      GO TO 35
34     RLONSTK=RLNSTKO
      RLATSTK=RLTSTKO
      RUDPEU=RPEDO
      THROT=106.
35    CONTINUE
       GO TO 110
30     CONTINUE
      IF(TIME.GE..1)GO TO 301
      ISWWAT=0
      ISWATC=0
      ISWATCP=0
301   CONTINUE
      XE2=XE*CPSI+YE*SPSI
      VXCOM=(VXCOMO+VERI)*(XE2/XCOMO)
      VERI=VERI+.2*(VXCOM-VX)*DT*RK2V
      IF(ABS(XE2).LT.10.)IXSW=1
      IF(IXSW.EQ.0)XCOM=XE2
      IF(IXSW.EQ.0)XCOM1=XCOM
      IF(IXSW.EQ.1)XCOM1=XCOM1*.97
```

```
IF(IYSW.EQ.1)XCOM=XCOM1
XE1=XE2-XCOM
XEIL(1)=XE1
YE2=YE*CPSI-XE*SPSI
YCOM=(YCOMO+20.)*(XE2/XCOMO)-20.
IF(IYSW.EQ.0)YCOM1=YCOM
IF(IYSW.EQ.1)YCOM1=YCOM1*.97
 IF(IYSW.EQ.1)YCOM=YCOM1
IF(ABS(XE2).LT.10..AND.ABS(VX).LT.2.)IYSW=1
YE1=YE2-YCOM
YEIL(1)=YE1
 THETO=THETIC/57.3
IF(IXSW.EQ.1)RKTHTS=1.02*RKTHTS
IF(RKTHTS.GE.RKTHT6)RKTHTS=RKTHT6
RKVX=RKTHTS*THX
VXL(1)=VX-VXCOM
THETRC=RKTHTS*XEIL(NLAG(13))+RKVX*VXL(NLAG(14))+THETO
IF(IYSW.EQ.1)RKYPHI=RKYPHI*1.02
IF(RKYPHI.LT.RKYPMAX)RKYPHI=RKYPMAX
RKPHIVY=THY*RKYPHI
VYL(1)=VY
PHIC=RKYPHI*YEIL(NLAG(1))+RKPHIVY*VYL(NLAG(2))+PHIIC/57.3
 IF(ABS(PHIC).GT..3)PHIC=SIGN(.3,PHIC)
 IF(THETHC.GT..3)THETHC=.3
 IF(THETHC.LT.0.)THETRC=0.
THETEL(1)=THETR
THER=THETEL(NLAG(3))
QBL(1)=QB
RLONST1=RLNTHT2*(THER-THETRC)+RLNQB2*QBL(NLAG(4))
IF(ABS(RLONST1-RLN1OLD).LT.THRESH1) RLONST1=RLN1OLD
IF(RLONST1.NE.RLN1OLD)RLNAF=RLNAF+1.
RLNVF1=(RLONST1-RLN1OLD)/DT
RLNVF2=ABS(RLNVF1)
RLNVF3=RLNVF3+RLNVF2
RLNVF=RLNVF3/(TIME+.05)
SGNLNV=SIGN(1.,RLNVF1)
IF(ABS(RLNVF1).LT..5)SGNLNV=0.
IF((SGNLNV+SGNLNVO).EQ.0..AND.SGNLNV.NE.0.)RLNSC1=RLNSC1+1.
RLNSC=RLNSC1/(TIME+.05)
SGNLNVO=SGNLNV
RLN1OLD=RLONST1
RLONST2=EXPZLN*(RLONST2-RLONST1)+RLONST1
IF(RLONST2.LT.(-7.5-RLNSTKO))RLONST2=-7.5-RLNSTKO
IF(RLONST2.GT.(3.7-RLNSTKO))RLONST2=3.7-RLNSTKO
RLONSTK=RLONST2+RLNSTKO
PHIEL(1)=PHIR
PBL(1)=PB
RLAT1=RLTPHI2*(PHIEL(NLAG(5))-PHIC)+RLTPB2*PBL(NLAG(6))
IF(ABS(RLAT1-RLT1OLD).LT.THRESH2) RLAT1=RLT1OLD
IF(RLAT1.NE.RLT1OLD)RLTAF=RLTAF+1.
RLATVF1=(RLAT1-RLT1OLD)/DT
RLTVF2=ABS(RLATVF1)
RLTVF3=RLTVF3+RLTVF2
RLTVF=RLTVF3/(TIME+.05)
SGNLTV=SIGN(1.,RLATVF1)
```

```
      IF(ABS(RLATVF1).LT..2)SGNLTV=0.
      IF((SGNLTV+SGNLTVO).EQ.0..AND.SGNLTV.NE.0.) RLTSC1=RLTSC1+1.
      RLTSC=RLTSC1/(TIME+.05)
      SGNLTVO=SGNLTV
      RLT1OLD=RLAT1
      RLAT2=EXP7LT*(RLAT2-RLAT1)+RLAT1
      RLATSTK=RLAT2+RLTSTKO
      RHL(1)=RH
      PSIC=PSIW
      IF(ABS(PSIC).GT.1.)PSIC=SIGN(1.,PSIC)
      PSICOM=PSIC*57.3
      PSIEL(1)=PSIP-PSIC
      IF(ABS(PSIEL(1)).GT..3)PSIEL(1)=SIGN(.3,PSIEL(1))
      RUDP1=RPHW1*RHL(NLAG(7))+RPPSI1*PSIEL(NLAG(8))
      BETA1=BETA
      IF(ABS(BETA1).GT.30.)BETA1=SIGN(30.,BETA1)
      RUDP1=RUDP1+RKRUD2*BETA1
      IF(ABS(RUDP1-RUDOLD).LT.THRESH3)RUDP1=RUDOLD
      IF(RUDP1.NE.RUDOLD)PEDAF=PEDAF+1.
      PEDVF1=(RUDP1-RUDOLD)/DT
      PEDVF2=ABS(PEDVF1)
      PEDVF3=PEDVF3+PEDVF2
      PEDVF=PEDVF3/(TIME+.05)
      SGNPV=SIGN(1.,PEDVF)
      IF(ABS(PEDVF1).LT.2.)SGNPV=0.
      IF((SGNPV+SGNPVO).EQ.0..AND.SGNPV.NE.0.)PEDSC1=PEDSC1+1.
      PEDSC=PEDSC/(TIME+.05)
      SGNPVO=SGNPV
      RUDOLD=RUDP1
      RUDP2=EXP7R*(RUDP2-RUDP1)+RUDP1
      RUDPED=RUDP2+RPEDO
       PHICOM=PHIC*57.3
       THETCOM=THETHC*57.3
      ALTCOM=ALT
      IF(ABS(XE).LT.10..AND.ABS(YE).LT.10.)IYZ=1
      IF(IYZ.EQ.0)ALTCOM=HIC
      IF(ABS(ALT-HTDO-HHOV).LT..5)IHSW=1
      IF(IHSW.EQ.1)ALTCOM=HTDO+HHOV
      ALTDCOM=0.
      IF(IYZ.EQ.1)ALTDCOM=-5.*(ALT-HTDO-HHOV)/(HIC-
     1   HTDO-HHOV)
      IF(IHTRACK.EQ.1)ALTCOM=HTDO+HHOV
       ALTE=ALT-ALTCOM
       IF(ABS(ALTE).GT.10.)ALTE=SIGN(10.,ALTE)
       ALTEI=ALTEI+DT*ALTE
       IF(ABS(ALTEI).GT.100.)ALTEI=SIGN(100.,ALTEI)
      IF(IHTRACK.EQ.1) ALTDCOM=HDTDKE(1)
      ALTDE=ALTD-ALTDCOM
       IF(ABS(ALTDE).GT. 5.) ALTDE=SIGN( 5.,ALTDE)
      ADECK=(HDTDKE(1)-VDECKP)/DT
       VDECKP=HDTDKE(1)
      ALTDDE=-VDD-ADECK
      IF(ABS(ALTDDE).GT.5.)ALTDDE=SIGN(5.,ALTDDE)
      ALTEL(1)=ALTE
      ALTEIL(1)=ALTEI
```

```
      ALTDEL(1)=ALTDE
      ALTDDEL(1)=ALTDDE
      IF(TTHROT.LT.TDEL3) GO TO 37
      THROT1=RTHROT*ALTEL(NLAG(9))+RTHROT1*ALTDEL(NLAG(10))
     1+RTHROT2*ALTFIL(NLAG(11))+RTHROT3*ALTDDEL(NLAG(12))
      IF(THROT1.GT.THLIM1)THROT1=THLIM1
      IF(THROT1.LT.THLIM3)THROT1=THLIM3
      IF(THROT1.GT.THLIM2.AND.ISWWAT.EQ.0)THROT1=THLIM2
      IF(THROT.GT.101..AND.ISWWAT.EQ.0)ISWATC=1
      IF(THROT.LT.99..AND.ISWWAT.EQ.1)ISWATC=0
      IF(ISWATC.NE.ISWATCP)TTHROT=0.
37    CONTINUE
      IF(TTHROT.LT.TDEL1)GO TO 38
      ISWWAT=ISWATC
      IF(TTHROT.LT.TDEL3)GO TO 38
      IF(ABS(THROT1-THROTO).LT.THRESH4)THROT1=THROTO
      IF(THROT1.NE.THROTO)TAF=TAF+1.
      TVF1=(THROT1-THROTO)/DT
      TVF2=ABS(TVF1)
      TVF3=TVF3+TVF2
      TVF=TVF3/(TIME+.05)
      SGNTV=SIGN(1.,TVF1)
      IF(ABS(TVF1).LT.2.)SGNTV=0.
      IF((SGNTV+SGNTVO).EQ.0..AND.SGNTV.NE.0.)TSC=TSC+1.
      TSCPSEC=TSC/(TIME+.05)
      SGNTVO=SGNTV
      THROTO=THROT1
      THROT2=EXPZT*(THROT2-THROT1)+THROT1
      THROT=THROTIC+THROT2
       IF(THROT.LT.THRTMIN) THROT=THRTMIN
38    TTHROT=TTHROT+DT
      ISWATCP=ISWATC
      THTNC=THTNIC
      DO 36 I=1,19
      THETEL(21-I)=THETEL(20-I)
      QAL(21-I)=QAL(20-I)
      XFIL(21-I)=XFIL(20-I)
      YFIL(21-I)=YFIL(20-I)
      VXL(21-I)=VXL(20-I)
      VYL(21-I)=VYL(20-I)
      PHIEL(21-I)=PHIEL(20-I)
      PHL(21-I)=PHL(20-I)
      PSIEL(21-I)=PSIEL(20-I)
      RAL(21-I)=RAL(20-I)
      ALTEL(21-I)=ALTEL(20-I)
      ALTDEL(21-I)=ALTDEL(20-I)
      ALTFIL(21-I)=ALTFIL(20-I)
      ALTDDEL(21-I)=ALTDDEL(20-I)
36    CONTINUE
       GO TO 110
40     THTNC=A1.
       THROT=100.+6.*SIN(5.*TIME)
       ISWWAT=1
       TMAX=13.
       GO TO 110
```

```
60    RMACH=.3
 50   THROT=THROTIC
      ISWHAT=1
      RLONSTK=RLNSTKO
      RLATSTK=RLTSTKO
      RUDPED=RPEDO
      THTNC=THTNIC
      RINPUT=0.
      IF(NCHK.EQ.1) GO TO 51
      IF(NCHK.EQ.2) GO TO 52
      IF(NCHK.EQ.3) GO TO 53
      IF(NCHK.EQ.4) GO TO 54
      GO TO 55
51    RINPUT=RMAG
      IF(TIME.LT.T0) RINPUT=0.
      GO TO 55
52    RINPUT=RMAG
      IF(TIME.LT.T0) RINPUT=0.
      IF(TIME.GT.T1) RINPUT=0.
      GO TO 55
53    IF(TIME.LT.T1) RINPUT=RMAG
      IF(TIME.LT.T0) RINPUT=0.
      IF(TIME.GT.T1) RINPUT=-RMAG
      IF(TIME.GT.T2) RINPUT=0.
      GO TO 55
54    RINPUT=RMAG*SIN(FRECK*(TIME-T0))
      IF(TIME.LT.T0) RINPUT=0.
55    CONTINUE
      IF(NCHK.EQ.1) THROT=THROTIC+RINPUT
      IF(NCHK.EQ.2) RLONSTK=RLNSTKO+RINPUT
      IF(NCHK.EQ.3) RLATSTK=RLTSTKO+RINPUT
      IF(NCHK.EQ.4) THTNC=THTNIC+RINPUT
      IF(NCHK.EQ.5) RUDPED=RPEDO+RINPUT
110   CONTINUE
      IF(ABS(RUDPED).GT.2.)RUDPED=SIGN(2.,RUDPED)
     STABM=STAB-RSAS
     IF(IZOOF.EQ.1) RLONSTK=TAB1(STABM,0.,0.,75,1)
      IF(RLONSTK.GE.3.7)RLONSTK=3.7
      IF(RLONSTK.LE.-7.5)RLONSTK=-7.5
      IF(ABS(RLATSTK).GE.4.)RLATSTK=SIGN(4.,RLATSTK)
      IF(THROT.GT.106.) THROT=106.
     IF(THROT.LT.60.) THROT=60.
100   CONTINUE
      IF(IROOF)3,5,5
3     DO 4 I=1,20
      THRTFL(I)=0.
      CAL(I)=0.
      XFIL(I)=XE2-XCOMO
      YFIL(I)=YE1-YCOMO
      VVL(I)=VY
      PHTFL(I)=0.
      PAL(I)=PH
      PSTFL(I)=0.
      RHL(I)=RH
      ALTFL(I)=0.
```

```
      ALTDEL(I)=0.
      ALTEIL(I)=0.
      ALTDDEL(I)=0.
      VXL(I)=VX-VXCOMO
4     CONTINUE
5     CONTINUE
        RETURN
      END
```

****

```
     11301111    STABILATOR VS STICK
-7.5    -6.6    -5.5    -4.35     -3.1  -2.05   -1.7     -.5     .6    01
-10.    -8.     -6.     -4.      -2.     -.5     0.      2.     4.     02
1.6     2.5     3.3     3.7                                            03
6.      8.      10.     11.3                                           04
     20201111   RUDDER VS PEDAL
 -2.1   2.1                                                            01
 -15.   15.                                                            02
     30701111    AILERON VS LATERAL STICK
 -4.    -2.5    -2.066  0.      2.066   2.5     4.                     01
-12.    -7.5    -6.2    0.      6.2     7.5     12.                    02
     40401111     FRON PITCH JET AREA VS ATICK
-7.5    -3.9    -.5     3.7                                            1
7.23    7.23    0.      0.                                             2
     50701111    REAR PITCH JET AREA VS STABILATOR
 -10.  2.       4.      6.      8.      10.     11.3                   1
 0.     0.      2.17       4.34  6.51  8.68    8.68                    2
     60701111    ROLL JET AREA VS AILERON
-12.    -7.5    -6.2    0.      6.2     7.5     12.                    1
4.2        4.2  3.696   0.      0.      .54     4.2                    2
     70501111    YAW JET AREA VS RUDDER
-15.    -10.    0.      10.     15.                                    01
3.5     3.5     0.      3.5     3.5                                    2
     80401111    DUCT PRESSURE RATIO VS NOZZLE TOTAL AREA
0.      2.      4.      6.      8.      10.     12.     14.     20.5   1
1.      .976    .968    .963    .959    .955    .95     .946    .5     2
     90501111    FRONT PITCH JET FORCE/MAX VS AREA
0.      2.      4.      6.      7.4                                    1
0.      .2712   .593    .68     1.                                     2
    100601111   REAR PITCH JET FUNCTION VS AREA
0.      2.      4.      6.      7.4     9.                             1
0.      .333    .598    .793    .897  1.                               2
    110501111   YAW JET FUNCTION
0.      1.      2.      3.      3.5                                    1
0.      .1428   .51428  .686    1.                                     2
    120501111   UP THRYST ROLL JET FUNCTION
0.      2.      3.      4.      4.2                                    1
0.      .53     .79     .973    1.                                     2
    130601111   DOWN THRUST ROLL JET FUNCTION
 0.     .8      1.8     2.8     3.8     4.2                            1
0.      .09     .19     .275    .36     .388                           02
    141001111       ENGINE TIME CONSTANT VS RPM
```

```
        0.      56.     6A.     75.     81.5    87.     91.5    95.5    99.6    1
        3.25    1.1     .55     .4      .278    .2      .17     .15     .15     2
        106.                                                                    3
        .15                                                                     4
           151404111  GROSS THRUST INCREASE DUE TO FORWARD SPEED
        0.      25.     30.     40.     50.     60.     70.     80.     90.     1
                                                                                2
     .1 0.      14.7    19.3    33.49   54.     79.52   128.68  163.    225.    3
    .2  0.      55.7    74.     131.    208.    306.    493.    626.    865.    4
    .3  0.      97.     129.    194.    362.    531.    860.    1088.   1503.   5
        95.     100.    102.    104.    106.                                    6
                                                                                7
    .1  245.    319.    343.    352.    367.                                    8
    .2  939.    1041.   1209.   1340.   1443.                                   9
    .3  1433.   1921.   2114.   2257.   2690.                                   10
           161403111  FRONT MAIN NOZZLE STATIC THRUST
        0.      25.     30.     40.     50.     60.     70.     80.     90.     1
        0.      706.    921.    1514.   2269.   3116.8  4144.   5502.   7000.   2
    9.  0.      594.    808.    1401.6  2156.   3004.1  4031.   5390.   6887.   3
    18. 0.      0.      0.      603.6   1358.   2206.   3233.   4591.   6089.   4
        95.     100.    102.    104.    106.                                    5
        8047.   8885.   9169.   9388.   9462.                                   6
    9.  7933.   8772.   9069.   9341.   9367.                                   7
    18. 7135.   8087.   8442.   9020.   9242.                                   8
           171403111    REAR NOZZLE STATIC THRUST
        0.      25.     30.     40.     50.     60.     70.     80.     90.     1
    0.  0.      125.7   188.7   428.    839.    1434.   2295.   3823.   5878.   2
    9.  0.      0.      0.      0.      369.8   965.    1826.   3353.   5409.   3
    18.                                                 418.    1945.   4001.   4
        95.     100.    102.    104.    106.                                    5
        7410.   8433.   9519.   9979.   10167.                                  6
    9.  6940.   8468.   9067.   9622.   9679.                                   7
    18. 5532.   6479.   7621.   8714.   9168.                                   8
           181304222  RAM DRAG VS N1 AND MACH
        0.      28.8    40.     60.     80.     85.     87.5    89.     94.     1
                                                                                2
    .1  0.      201.    313.    546.    855.    944.    1001.   1038.   1158.   3
    .2  0.      540.    751.    1243.   1855.   2027.   2146.   2215.   2432.   4
    .3  304.    1132.   1454.   2252.   2900.   3152.   3555.   3644.   3907.   5
        100.    102.    104.    106.                                            6
                                                                                7
    .1  1249.   1346.   1394.   1442.                                           8
    .2  2688.   2764.   2859.   2946.                                           9
    .3  4217.   4322.   4426.   4531.                                           10
           190701111  CY VS BETA M#.4 POWER OFF
        -30.    -15.    -6.     0.      6.      15.     30.                     1
    6.  .435    .225    .09     0.      -.09    -.225   -.435                   2
           200503111  CY VS BETA THETJ#60..VJ/VO#5.4
    60. -20.    -10.    0.      10.     20.                                     1
    0.  .4      .22     0.      -.24    -.4                                     2
    10. .35     .18     0.      -.18    -.32                                    3
    20. .3      .15     0.      -.15    -.3                                     4
           210503111  CY VS BETA.ALPHA.THETJ#75.VJ/VO#5.4
    75. -20.    -10.    0.      10.     20.                                     1
    0.  .37     .21     0.      -.25    -.49                                    2
```

```
10.    .35   .21   0.     -.16   -.3                                        3
20.    .31   .15   0.     -.15   -.31                                       4
       220503111   CY VS BETA ALPHA,THETJ#60.,VJ/VO#11.9
60.    -20.   -10.   0.    10.    20.                                       1
0.     .3    .17   0.     -.2    -.4                                        2
5.     .36   .18   0.     -.18   -.36                                       3
10.    .36   .18   0.     -.16   -.32                                       4
       230503111   CY VS BETA,ALPHA,THETJ#90.,VJ/VO#11.9
90.    -20.  -10.  0.     10.    20.                                        1
0.     .4    .23   0.     -.18   -.35                                       2
10.    .38   .17   0.     -.18   -.31                                       3
20.    .27   .135  0.     -.135  -.27                                       4
       240504211   CROLL VS BETA,ALFA,THTJ=0.,VJ/VO=0.,M=.4
0.     -2.    0.     2.      4.      6.                                     1
0.     .002   0.     -.0004 -.0007 -.0013                                   2
4.     .002   0.     -.001  -.002  -.0035                                   3
8.     .0015  0.     -.0001 -.0015 -.0028                                   4
12.    .0014  0.     -.0018 -.0038 -.0059                                   5
       250407111   CROLL VS BETA,ALPHA,THTJ#60.,VJ/VO#5.4
60.    -40.   -30.  -20.   -10.   0.    10.     20.     30.     40.         1
-10.   -.005  -.003 -.002  -.001  0.    .001    .002    .003    .004        2
-5.    .026   .02   .013   .006   0.    -.006   -.012   -.02    -.026       3
0.     .094   .074  .052   .031   0.    -.012   -.022   -.054   -.074       4
5.     .114   .084  .056   .038   0.    -.026   -.055   -.084   -.11        5
10.    .12    .098  .072   .04    0.    -.046   -.09    -.138   -.172       6
15.    .13    .102  .074   .042   0.    -.054   -.102   -.142   -.178       7
20.    .13    .098  .066   .034   0.    -.034   -.074   -.116   -.164       8
       260407111   CROLL CS BETA,ALPHA,THTJ#75.,VJ/VO#5.4
75.    -40.   -30.  -20.   -10.   0.    10.     20.     30.     40.         1
-10.   .018   .021  .015   .008   0.    -.009   -.019   -.028   -.036       2
-5.    .027   .02   .014   .007   0.    -.007   -.014   -.02    -.026       3
0.     .065   .052  .038   .02    0.    -.026   -.056   -.086   -.114       4
5.     .1     .098  .084   .054   0.    -.036   -.056   -.06    -.056       5
10.    .148   .114  .08    .04    0.    -.05    -.1     -.14    -.178       6
15.    .129   .107  .083   .047   0.    -.075   -.127   -.155   -.175       7
20.    .12    .094  .06    .03    0.    -.043   -.084   -.12    -.152       8
       270407111   CROLL VS ALPHA,BETA,THETJ#60.,VJ/VO#11.9
60.    -40.   -30.  -20.   -10.   0.    10.     20.     30.     40.         1
-10.   -.033  -.025 -.019  -.01   0.    .005    .01     .015    .02         2
-5.    -.01   -.008 -.006  -.003  0.    .002    .004    .008    .01         3
0.     .021   .026  .022   .014   0.    -.019   -.037   -.054   -.068       4
5.     .048   .045  .04    .028   0.    -.05    -.084   -.102   -.12        5
10.    .112   .109  .082   .042   0.    -.056   -.076   -.114   -.144       6
15.    .134   .118  .08    .046   0.    -.064   -.12    -.156   -.204       7
20.    .17    .13   .087   .04    0.    -.04    -.07    -.118   -.15        8
       280408111   CROLL VS ALPHA,BETA,THETJ#90.,VJ/VO#11.9
90.    -40.   -30.  -20.   -10.   0.    10.     20.     30.     40.         1
-10.   0.                                                                   2
-5.    .02    .015  .01    .005   0.    -.006   -.01    -.018   -.024       3
0.     .033   .025  .016   .008   0.    -.009   -.015   -.024   -.032       4
5.     .042   .032  .022   .012   0.    -.01    -.022   -.032   -.042       5
10.    .111   .083  .057   .029   0.    -.029   -.058   -.085   -.112       6
15.    .107   .094  .079   .052   0.    -.03    -.12    -.168   -.204       7
       290504211   CN VS BETA,ALPHA,THETJ=0.,VJ/VO=0.
0.     -2.    0.    2.     4.     6.                                        1
```

```
 0.    -.0087  0.     .0075  .0155  .024                                  2
 4.    -.008   0.     .0065  .015   .024                                  3
 8.    -.007   0.     .0055  .0125  .022                                  4
12.    -.005   0.     .005   .011   .0205                                 5
       300507111   CN VS ALPHA,BETA,THETJ#60.,VJ/VO#5.4
60.    -20.    -10.    0.     10.    20.                                  1
-10.   -.094   -.048   0.     .046   .093                                 2
-5.    -.12    -.064   0.     .06    .12                                  3
0.     -.07    -.052   0.     .116   .154                                 4
5.     -.085   -.06    0.     .086   .15                                  5
10.    -.102   -.058   0.     .082   .112                                 6
15.    -.084   -.048   0.     .046   .084                                 7
20.    -.068   -.03    0.     .022   .044                                 8
       310507111   CN VS ALPHA,BETA,THETJ#75,VJ/VO#5.4
75.    -20.    -10.    0.     10.    20.                                  1
-10.   -.152   -.088   0.     .084   .134                                 2
-5.    -.104   -.058   0.     .082   .134                                 3
0.     -.115   -.065   0.     .07    .12                                  4
5.     -.09    -.065   0.     .087   .145                                 5
10.    -.084   -.048   0.     .052   .091                                 6
15.    -.085   -.044   0.     .04    .08                                  7
20.    -.054   -.026   0.     .022   .038                                 8
       320507111   CN VS ALPHA,BETA,THETJ#75,VJ/VO#11.9
60.    -20.    -10.    0.     10.    20.                                  1
-10.   -.075   -.044   0.     .062   .124                                 2
-5.    -.09    -.06    0.     .12    .138                                 3
0.     -.094   -.052   0.     .06    .112                                 4
5.     -.057   -.046   0.     .106   .142                                 5
10.    -.076   -.044   0.     .05    .09                                  6
15.    -.06    -.032   0.     .036   .07                                  7
20.    -.032   -.016   0.     .016   .03                                  8
       330507111   CN VS ALPHA,BETA,THTJ#90,VJ/VO#11.9
90.    -20.    -10.    0.     10.    20.                                  1
-10.   -.108   -.068   0.     .07    .12                                  2
-5.    -.096   -.07    0.     .17    .21                                  3
0.     -.085   -.05    0.     .16    .2                                   4
5.     -.078   -.056   0.     .072   .122                                 5
10.    -.092   -.054   0.     .046   .086                                 6
15.    -.05    -.032   0.     .058   .118                                 7
20.    -.042   -.02    0.     .018   .034                                 8
       340501111   CLDRVS SPEED IN KNOTS
       40.     80.    120.    160.   200.                                 1
       .00018 .00017.00014  .00012 .0001                                  2
       350201111   CNDA VS ALPHA
       2.      12.                                                        1
       .00025-.00035                                                      2
       360201111   CLR VS ALPHA
       6.      12.                                                        1
       .143    .212                                                       2
       370702121   CL VS ALPHA,DM,VJ#0,THTJ#0
       -5.     0.     5.      10.    12.     15.     20.                  1
0.     .004    .3     .53     .83    .94     1.05    1.1                  2
6.     .05     .35    .62     .89    .99     1.09    1.13                 3
       380802121        CL VS ALPHA,DM,VJ/VO=5.4,THTJ30.
30.    -5.     0.     5.      8.     10.     12.     15.     20.          1
```

```
  6.   -.12   .12    .37    .53    .63    .74    .87    .95        2
 12.   -.04   .21    .46    .61    .71    .82    .94    1.01       3
       390802121 CL VS ALPHA,DH,VJ#5.4,THTJ#45
 45.   -5.    0.     5.     8.     10.    12.    15.    20.        1
  6.   -.22   .04    .29    .44    .54    .64    .77    .975       2
 12.   -.15   .1     .35    .5     .6     .7     .83    1.03       3
       400802121   CL VS ALPHA,DH,VJ/VO=5.4,THTJ=60
 60.   -5.    0.     5.     8.     10.    12.    15.    20.        1
  6.   -.3    -.05   .21    .36    .46    .55    .67    .88        2
 12.   -.26   0.     .26    .4     .5     .59    .72    .94        3
       410802121  CL VS ALPHA,DH,VJ/VO#5.4,THTJ#75
 75.   -5.    0.     5.     8.     10.    12.    15.    20.        1
  6.   -.43   -.18   .07    .21    .31    .41    .55    .79        2
 12.   -.35   -.11   .13    .28    .38    .47    .62    .86        3
       420802121   CL VS ALPHA,DH,VJ/VO#5.4,THTJ#90.
 90.   -5.    0.     5.     8.     10.    12.    15.    20.        1
  6.   -.62   -.25   .01    .16    .26    .36    .51    .7         2
 12.   -.43   -.2    .06    .21    .31    .42    .56    .81        3
       430802121  CL VS ALPHA,DH,VJ/VO#11,THETJ#45
 45.   -5.    0.     5.     8.     10.    12.    15.    20.        1
  6.   -.55   -.3    -.03   .12    .23    .33    .46    .6         2
 12.   -.5    -.22   .04    .19    .3     .4     .54    .74        3
       440802121  CL VS ALPHA,DH,VJ/VO#11,THETJ#60
 60.   -5.    0.     5.     8.     10.    12.    15.    20.        1
  6.   -.66   -.36   -.1    .04    .14    .23    .33    .45        2
 12.   -.59   -.3    -.05   .1     .2     .3     .42    .54        3
       450802121   CL VS ALPHA,DH,VJ/VO#11,THTJ#75
 75.   -5.    0.     5.     8.     10.    12.    15.    20.        1
  6.   -.8    -.59   -.29   -.14   -.05   .04    .16    .36        2
 12.   -.74   -.46   -.21   -.08   0.     .08    .17    .36        3
       460702121  CD VS ALPHA,DH,VJ/VO#5.4,THETJ#0
  0.   -5.    0.     5.     10.    12.    15.    20.               1
  0.   .106   .122   .146   .207   .262   .332   .528              2
  6.   .12    .123   .153   .217   .263   .36    .556              3
       470801111  CD VS ALPHA,DH,VJ/VO#5.4,THTJ#30
 30.   -5.    0.     5.     8.     10.    12.    15.    20.        1
 12.   .147   .159   .182   .214   .242   .28    .34    .41        2
       480802121   CD ALPHA,DH,VJ/VU=5.,THTJ=45.
 45.   -5.    0.     5.     8.     10.    12.    15.    20.        1
  6.   .161   .151   .18    .196   .233   .268   .338   .505       2
 12.   .164   .16    .197   .224   .252   .294   .368   .526       3
       490802121  CD VS ALPHA,DH,VJ/VO=5.4,THTJ=60.
 60.   -5.    0.     5.     8.     10.    12.    15.    20.        1
  6.   .142   .156   .183   .208   .235   .262   .32    .42        2
 12.   .141   .155   .193   .213   .24    .272   .33    .45        3
       500802121  CD VS ALPHA,DH,VJ/VO)=5,THTJ=75
 75.   -5.    0.     5.     8.     10.    12.    15.    20.        1
  6.   .156   .14    .158   .18    .207   .232   .286   .4         2
 12.   .16    .15    .172   .20    .226   .25    .3     .408       3
       510802121  CD VS ALPHA,DH,VH  VJ/VO=5,THTJ=90
 90.   -5.    0.     5.     8.     10.    12.    15.    20.        1
  6.   .161   .148   .166   .188   .21    .232   .272   .372       2
 12.   .168   .156   .179   .206   .222   .254   .296   .368       3
       520802121   CD VS ALPHA,DH,VJ/VO=11,THTJ=45
 45.   -5.    0.     5.     8.     10.    12.    15.    20.        1
```

```
6.    .192    .145  .12    .144    .165  .19    .228   .273        2
12.   .195    .13   .126   .157    .183  .213   .252   .31         3
      530802121  CD VS ALPHA,DH,VJ/VO=11,THTJ=60
60.   -5.     0.    5.     8.      10.   12.    15.    20.         1
6.    .192    .12   .117   .137    .16   .186   .226   .346        2
12.   .188    .134  .134   .159    .176  .198   .236   .294        3
      540802121  CD VS ALPHA,DH,VJ/VO=11,THTJ=75
75.   -5.     0.    5.     8.      10.   12.    15.    20.         1
6.    .216    .1    .064   .072    .088  .108   .12    .178        2
12.   .246    .105  .074   .08     .095  .11    .134   .174        3
      550702121  CM VS ALPHA,DH,THTJ=0,VJ/VO=5.4
0.    -5.     0.    5.     10.     12.   15.    20.                1
0.    -.055   -.072 -.102  -.145   -.17  -.22   -.3                2
6.    -.14    -.165 -.19   -.24    -.27  -.32   -.39               3
      560802121  CM VXS ALPHA,DH,THTJ=30,VJ/VO=5.4
30.   -5.     0.    5.     8.      10.   12.    15.    20.         1
6.    -.055   -.024 -.03   -.034   -.038 -.042  -.036  -.018       2
12.   -.143   -.14  -.14   -.156   -.168 -.166  -.158  -.152       3
      570802121  CMVS,ALPHA,DH,THTJ=45,VJ/VO=5.4
45.   -5.     0.    5.     8.      10.   12.    15.    20.         1
6.    .007    -.005 .001   .003    .004  .004   .008   .018        2
12.   -.122   -.118 -.12   -.118   -.117 -.12   -.116  -.093       3
      580802121  CMVS ALPHA,DH,THTJ=60,VJ/VO=5.4
60.   -5.     0.    5.     8.      10.   12.    15.    20.         1
6.    .04     .038  .04    .037    .035  .034   .039   .076        2
12.   -.053   -.076 -.08   -.082   -.084 -.086  -.082  -.053       3
      590802121  CM VS ALPHA,DH,THTJ=75,VJ/VO=5.4
75.   -5.     0.    5.     8.      10.   12.    15.    20.         1
6.    .08     .079  .06    .045    .036  .026   .021   .02         2
12.   -.042   -.046 -.055  -.074   -.088 -.096  -.102  -.108       3
      600802121  CM VS ALPHA,DH,THTJ=90,VJ/VO=5.4
90.   -5.     0.    5.     8.      10.   12.    15.    20.         1
6.    .048    .08   .06    .04     .03   .024   .022   .018        2
12.   -.022   -.03  -.04   -.066   -.08  -.088  -.092  -.094       3
      610802121  CM VS ALPHA,DH,THTJ=45,VJ/VO=11.9
45.   -5.     0.    5.     8.      10.   12.    15.    20.         1
6.    .065    .059  .046   .048    .054  .064   .067   .04         2
12.   -.028   -.058 -.058  -.06    -.06  -.058  -.054  -.044       3
      620802121  CM VS ALPHA,DH,THTJ=60,VJ/VO=11.9
60.   -5.     0.    5.     8.      10.   12.    15.    20.         1
6.    .102    .057  .058   .062    .062  .059   .034   .018        2
12.   -.024   -.047 -.049  -.048   -.046 -.047  -.053  -.058       3
      630802121  CM VS ALPHA,DH,THTJ=75,VJ/VO=11.9
75.   -5.     0.    5.     8.      10.   12.    15.    20.         1
6.    .185    .085  .069   .066    .064  .06    .058   .05         2
12.   .044    -.02  -.032  -.032   -.034 -.04   -.053  -.08        3
      641101222  NOSE GEAR FORCE VS GEAR COMPRESSION
  -1.25  -1.16  -1.    -.83    -.77   -.513  -.33   -.17   0.      1
  -32000.-27000.-17600.-11200. -7600. -6347. -5000. -3000. 0.      2
  .01    1.                                                        3
  0.                                                               4
      650601222  MAIN FORCE VERSUS GEAR COMPRESSION
  -1.04  -1.    -.833  -.67   -.5833  -.0                          1
  -23000.-16000. -8000. -4300. 0.     0.                           2
      660201111  CNP VS ALPHA
```

```
 6.       12.                                                               1
 -.286    -.974                                                             2
   672201111      PITCH TIME HISTORY FDR# 38 AV-8A
 0.       1.25    1.75    2.      8.      8.5     9.      9.25    9.5       1
 5.5      5.      6.5     7.      7.      6.5     7.      6.5     6.4       2
 9.75     10.     10.25   10.5    11.     11.5    12.     12.5    12.75     3
 7.5      8.7     9.25    9.5     9.5     9.7     10.     9.5     9.2       4
 13.      13.25   15.     16.                                               5
 9.5      10.     10.     10.                                               6
   681201111      STEADY UHURBLE VERSUS RANGE
 -1838. -1800. -1700. -1600. -1400. -1200. -850.   -660.   -460.            1
  0.     -.04    -.06   -.056  -.03    -.04    -.097 -.12    -.14           2
 -300.    0.      100.                                                      3
                  .                                                         4
   6913 1111          STEADY WAURBLE VS RANGE
 -2700. -2500. -2250. -2000. -1500. -1100. -1000. -800.   -300.             1
  0.      .065    .07     .065    .035    0.      -.005   -.0175  0.        2
 400.     434.    510.    550.                                              3
 0.       .256    0.      0.                                                4
   700901111      DOWN WASH WITH ILBURB=1
 -2700. -2500. -2250. -2000. -1500. -1100. -1000. -800.   -600.             1
  0.      .065    .07     .065    .035    0.      -.005   -.0175  0.        2
   71 9 1111      SIGMA RANDOM BURBLE
 -2540. -1600. -1200. -1100. -1000. -800.   -600.   -440.    0.             1
  0.      .028    .035    .0355   .036    .0335   .021    0.      0.        2
   72 9 1111        TAU RANDOM HURBLE
 -2540. -1600. -1200. -1100. -1000. -800.   -600.   -440.    0.             1
  1.02    1.02    1.05    .95     .82     .5      .325    .25     .11       2
   73 7 5111      DCL 120 G. TANK MOD 4 ESTIMATE
 0.       .05     .1      .15     .2      .25     .3                        1
0.                                                                          2
30.      -.42    -.354   -.288   -.223   -.1356 -.108   -.05                3
50.      -.405   -.344   -.284   -.223   -.158  -.134   -.089               4
60.      -.4     -.341   -.282   -.223   -.175  -.167   -.14                5
98.      -.4     -.341   -.282   -.223   -.175  -.167   -.14                6
   743401111      STABILIZER VS TIME FDR#38
 0.       1.      1.25    1.5     1.75    2.      2.25    3.      4.        1
 6.6      6.5     6.5     7.      7.3     7.      6.5     6.5     6.1       2
 7.       7.05    7.2     7.4     7.6     7.75    9.      9.4     10.       3
 6.       6.5     5.      5.      6.5     6.      5.5     4.5     7.        4
 10.25    10.5    10.75   11.     11.5    11.75   12.     12.75   13.       5
 8.5      7.      6.7     7.1     7.      7.5     7.7     6.2     6.2       6
 14.      15.     16.     17.     18.     18.25   20.                       7
 6.5      7.      6.      5.7     5.7     5.      5.                        8
   751301111      STICK VS STABILIZER GEARING
 -10.     -8.     -6.     -4.     -2.     -.5     0.      2.      4.        1
 -7.5     -6.6    -5.55   -4.35   -3.1    -2.05   -1.7    -.5     .6        2
 6.       8.      10.     11.3                                              3
 1.6      2.5     3.3     3.7                                               4
   760505111      VX MEAN VSS=20 KNOT PSIS=0 Z=14.58
14.58    0.       37.     90.     125.    250.                              1
-18.75   17.702  17.414  21.578  23.11   25.38                              2
0.       15.359  17.308  21.199  23.39   25.23                              3
18.75    14.328  18.102  23.178  25.23   23.69                              4
54.      27.522  27.433  30.82   31.49   30.402                             5
```

104.	28.108	28.04	31.1	30.84	30.65	6
	770505111		VXBAR VSS=20 PSIS=0 Z=25			
25.	0.	37.	90.	125.	250.	1
-18.75	23.67	24.11	25.748	26.71	28.539	2
0.	19.729	19.932	23.777	25.29	26.06	3
18.75	21.278	21.646	25.986	26.29	25.11	4
54.	27.332	26.947	29.61	30.01	29.95	5
104.	30.46	30.46	30.46	30.19	30.03	6
	780505111		VXBAR VSS=20 PSI=0 Z=44.5			
44.58	0.	37.	90.	125.	250.	1
-18.75	27.80	29.32	29.29	29.81	29.0	2
0.	26.4	23.95	25.02	26.21	26.54	3
18.75	27.13	18.39	25.29	25.51	22.209	4
54.	30.28	29.95	29.65	30.11	28.96	5
104.	29.89	29.77	30.115	30.11	29.84	6
	790505111		VZBAR VXS=20 KNUTS PSI=0. Z=14			
14.58	0.	37.	90.	125.	250.	1
-18.75	-1.028	-1.34	-.656	-.824	-.709	2
0.	-1.228	-.6518	.2158	-.0588	-1.33	3
18.75	1.3136	1.2182	1.443	1.549	.333	4
54.	1.762	1.8173	1.153	2.17	1.407	5
104.	.174	.1488	-.853	-.1512	-.45238	6
	800505111		VYBAR VSS=20 PSI=0 Z=25			
25.	0.	37.	90.	125.	250.	1
-18.75	-2.002	-1.701	-1.604	-1.625	-.803	2
0.	-.7	-1.024	.3087	1.156	.118	3
18.75	1.8659	1.317	1.5036	1.287	.0797	4
54.	.98919	.8048	.821	2.039	1.	5
104.	.2	.15	.4736	1.307	.9506	6
	810505111		VYBAR VSS=20 PSU=0 Z=44.5			
44.58	0.	37.	90.	125.	250.	1
-18.75	-1.2212	-1.512	-1.548	-1.615	-.737	2
0.	2.024	.3314	.4089	.205	.374	3
18.75	1.848	6.7234	.1465	-.1	1.301	4
54.	-.268	1.0115	-1.225	-.1629	-.264	5
104.	-.4384	.0156	-1.574	-.539	-.856	6
	820505111		VZBAR VSS=20 PSI=0 Z=14.58			
14.58	0.	37.	90.	125.	250.	1
-18.75	-1.2	-.7189	-1.784	-.3395	.7242	2
0.	-3.192	-1.805	-1.21	-1.018	.25512	3
18.75	-.1856	.05179	1.541	1.199	1.9689	4
54.	1.2908	1.199	.2479	-.004	1.3	5
104.	.2201	.21219	.132	.4034	.778	6
	830505111		VZBAR VSS=20 PSI=0 Z=25			
25.	0.	37.	90.	125.	250.	1
-18.75	-.8612	-.475	-.965	.1313	.4674	2
0.	-1.03	-.437	.2829	-.222	.7198	3
18.75	-.3546	-.0464	-.1612	-1.199	1.061	4
54.	.5871	.6548	-.133	.365	.143	5
104.	0.	0.	.01215	-.0703	.633	6
	840505111		VZBAR VSS=20 PSI=0 Z=44			
44.58	0.	37.	90.	125.	250.	1
-18.75	-.6105	-.018	-.1286	.9025	1.011	2
0.	1.9	1.507	1.354	1.859	1.447	3
18.75	.8357	-8.792	.8027	1.256	-2.123	4

54.	2.668	1.307	3.044	1.092	2.1748	5
104.	2.207	2.064	1.6819	2.191	2.1758	6
	850505111		SIGX VSS=20 PSI=0 Z=14.58			
14.58	0.	37.	90.	125.	250.	1
-18.75	6.488	5.5158	4.719	4.549	3.8033	2
0.	4.5334	3.895	4.609	4.04	2.993	3
18.75	3.894	4.3837	4.257	3.968	3.026	4
54.	1.3287	1.3053	.9439	.6944	1.16	5
104.	.2948	.2999	.422	.3649	.2948	6
	860505111		SIGX VSS=20 SPI=0 Z=25.			
25.	0.	37.	90.	125.	250.	1
-18.75	4.0513	4.274	3.206	3.529	2.063	2
0.	3.828	3.92	4.46	2.68	2.409	3
18.75	4.339	3.436	4.25	3.964	2.703	4
54.	1.55	1.678	1.395	.9143	.79	5
104.	1.2	1.2	.9914	.64214	.3929	6
	870505111		SIGX VSS=20 PSI=0 Z=44			
44.58	0.	37.	90.	125.	250.	1
-18.75	2.779	3.196	1.94	1.5	1.57	2
0.	3.17	2.004	2.787	2.722	2.53	3
18.75	3.203	8.2835	3.595	3.411	6.015	4
54.	1.055	1.109	1.355	1.382	1.509	5
104.	.4958	.4877	.7277	.367	.515	6
	880505111		SIGY VSS=20 PSI=0 Z=14.58			
14.58	0.	37.	90.	125.	250.	1
-18.75	4.4409	4.2153	3.9148	3.72	3.028	2
0.	4.3388	4.7039	4.4203	4.45	3.279	3
18.75	3.3622	2.931	3.1038	3.47	2.4516	4
54.	1.4748	1.406	1.1226	.833	1.331	5
104.	.41318	.4135	.701	.7368	.2948	6
	890505111		SIGY VSS=20 PSI=0 Z=25			
25.	0.	37.	90.	125.	250.	1
-18.75	3.511	3.188	3.799	3.727	2.8192	2
0.	3.806	3.915	4.075	3.588	2.567	3
18.75	3.254	3.272	3.1415	3.277	2.5916	4
54.	1.3193	1.33091	1.741	1.206	.7909	5
104.	.413	.413	1.459	1.1274	.3929	6
	900505111		SIGY VSS=20 PSI=0 Z=44			
44.58	0.	37.	90.	125.	250.	1
-18.75	4.131	2.768	2.974	2.416	2.27	2
0.	4.444	3.8723	3.35	3.182	2.526	3
18.75	3.36	8.079	3.06	3.488	5.087	4
54.	1.798	1.5309	1.929	1.601	2.107	5
104.	.8585	.7434	1.259	.563	.7329	6
	910505111		SIG7 VSS=20 PSI=0 Z=14			
14.58	0.	37.	90.	125.	250.	1
-18.75	4.189	4.385	4.194	3.471	3.149	2
0.	3.42	3.193	3.376	3.156	2.7823	3
18.75	3.882	3.188	3.5457	2.848	2.373	4
54.	1.825	1.515	1.181	1.088	1.0456	5
104.	.43619	.36	.634	.556	.4175	6
	920505111		SIG7 VSS=20 PSI=0 Z=25			
25.	0.	37.	90.	125.	250.	1
-18.75	3.	3.256	3.1438	2.756	2.868	2
0.	3.037	2.866	3.358	2.667	2.226	3

18.75	3.399	2.61	2.7307	3.824	2.0169	4
54.	1.559	1.7179	2.0738	1.4359	.8847	5
104.	1.559	1.01	.4663	.6445	.64424	6
	930505111		SIGZ	VSS=20 PSI=0 Z=44		
44.58	0.	37.	90.	125.	250.	1
-18.75	3.425	2.623	2.296	2.115	1.719	2
0.	3.911	2.8529	2.757	2.737	2.136	3
18.75	4.202	11.04	2.718	3.138	8.5137	4
54.	1.803	1.782	2.168	1.716	2.006	5
104.	.6198	.595	.8391	.544	.6321	6

This page left intentionally blank.

APPENDIX B

PLOTS OF NON LINEAR TABLES

This page left intentionally blank.

TABLE NO.1

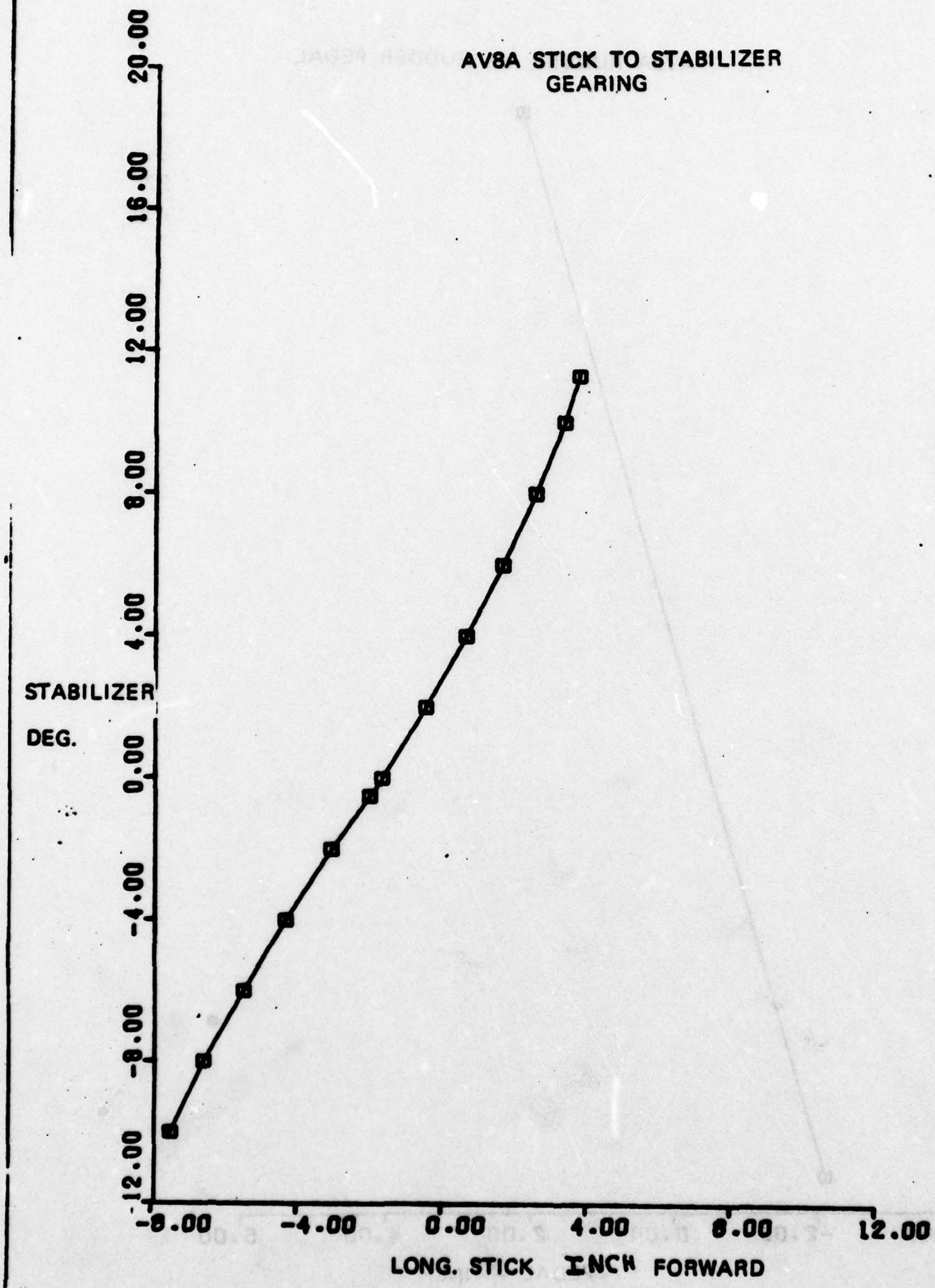

AV8A STICK TO STABILIZER
GEARING
STABILIZER
DEG.
20.00
16.00
12.00
8.00
4.00
0.00
-4.00
-8.00
-12.00
-8.00
-4.00
0.00
4.00
8.00
12.00
LONG. STICK INCH FORWARD

TABLE NO.2

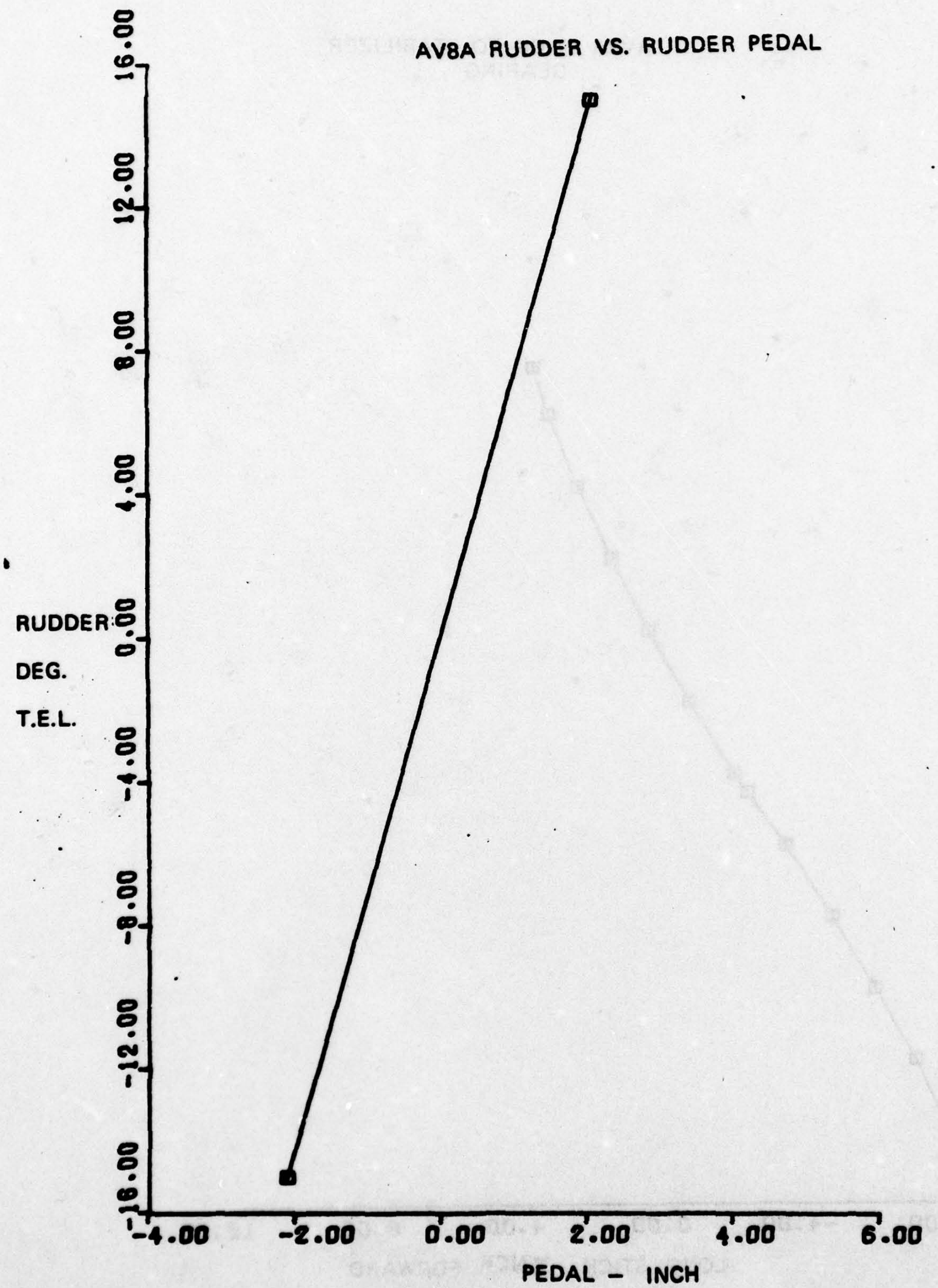

AV8A RUDDER VS. RUDDER PEDAL
RUDDER
DEG.
T.E.L.
16.00
12.00
8.00
4.00
0.00
-4.00
-8.00
-12.00
-16.00
-4.00
-2.00
0.00
2.00
4.00
6.00
PEDAL – INCH

TABLE NO 3

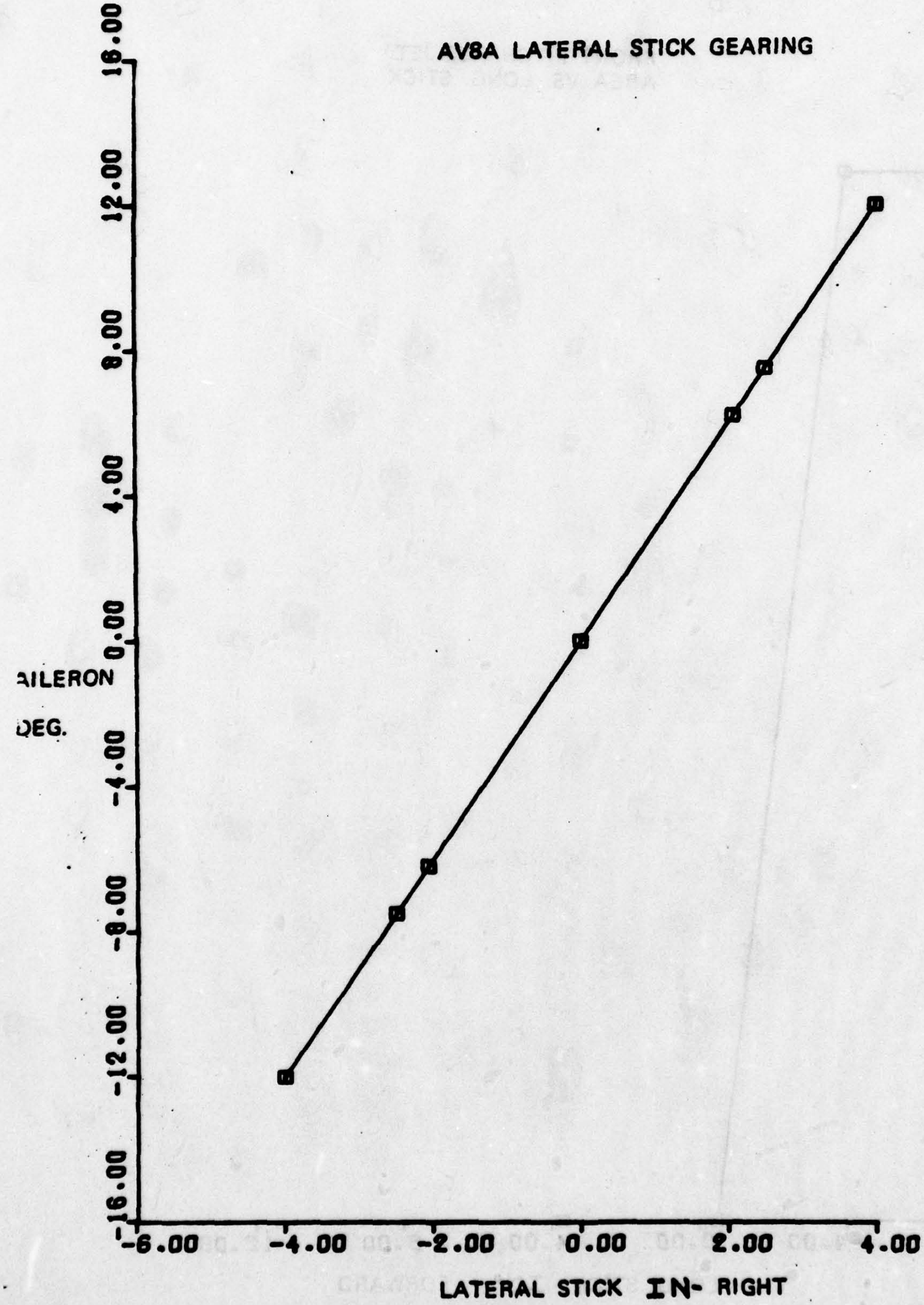

AV8A LATERAL STICK GEARING
16.00
12.00
8.00
4.00
0.00
-4.00
-8.00
-12.00
-16.00
AILERON
DEG.
-6.00
-4.00
-2.00
0.00
2.00
4.00
LATERAL STICK IN- RIGHT

TABLE NO.4

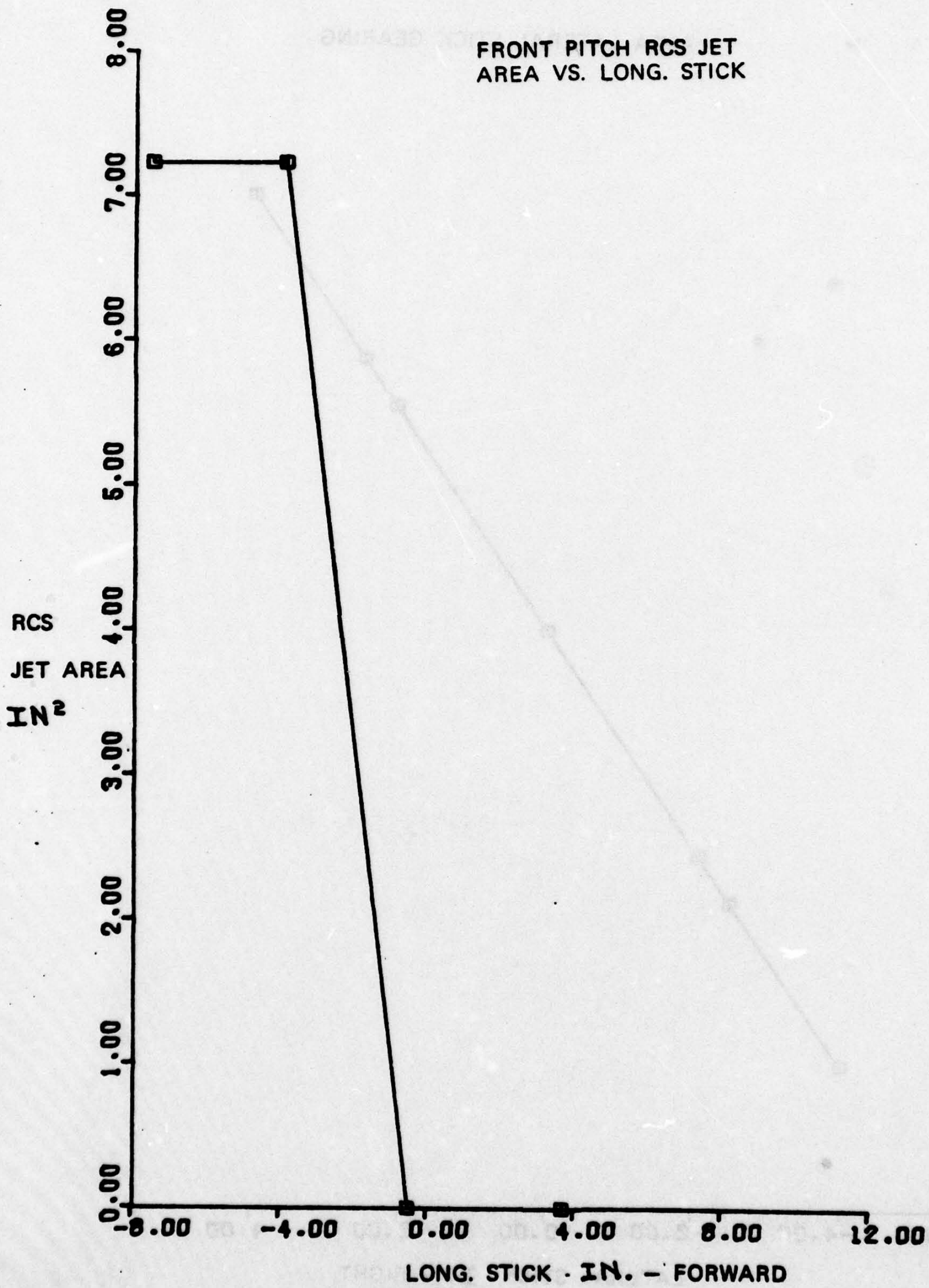

FRONT PITCH RCS JET
AREA VS. LONG. STICK
RCS
JET AREA
IN²
8.00
7.00
6.00
5.00
4.00
3.00
2.00
1.00
0.00
-8.00
-4.00
0.00
4.00
8.00
12.00
LONG. STICK - IN. - FORWARD

TABLE NO.5

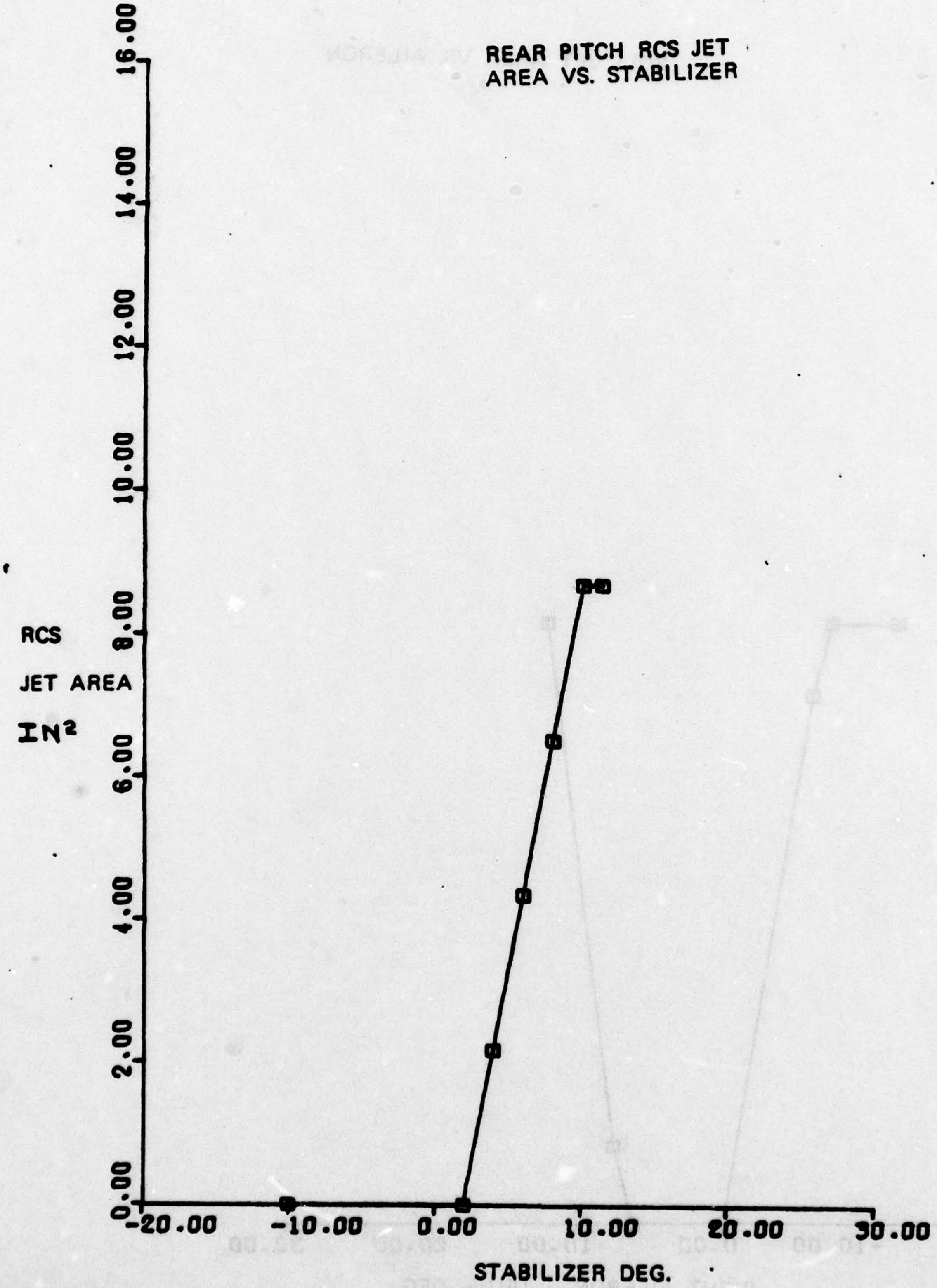

REAR PITCH RCS JET
AREA VS. STABILIZER
RCS
JET AREA
IN²
0.00
2.00
4.00
6.00
8.00
10.00
12.00
14.00
16.00
-20.00
-10.00
0.00
10.00
20.00
30.00
STABILIZER DEG.

TABLE NO.6

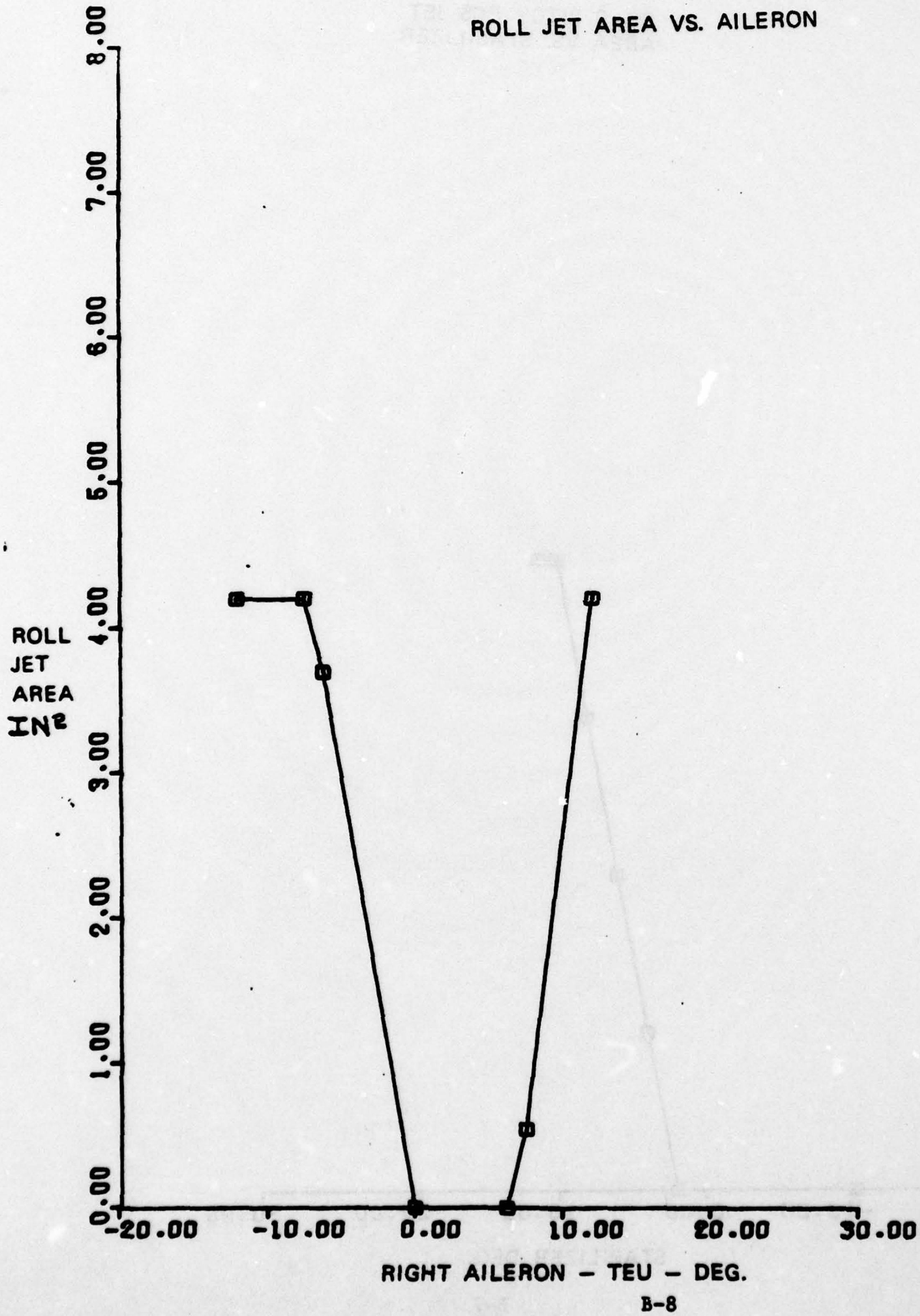

ROLL JET AREA VS. AILERON
ROLL JET AREA IN²
8.00
7.00
6.00
5.00
4.00
3.00
2.00
1.00
0.00
-20.00
-10.00
0.00
10.00
20.00
30.00
RIGHT AILERON – TEU – DEG.

TABLE NO.7

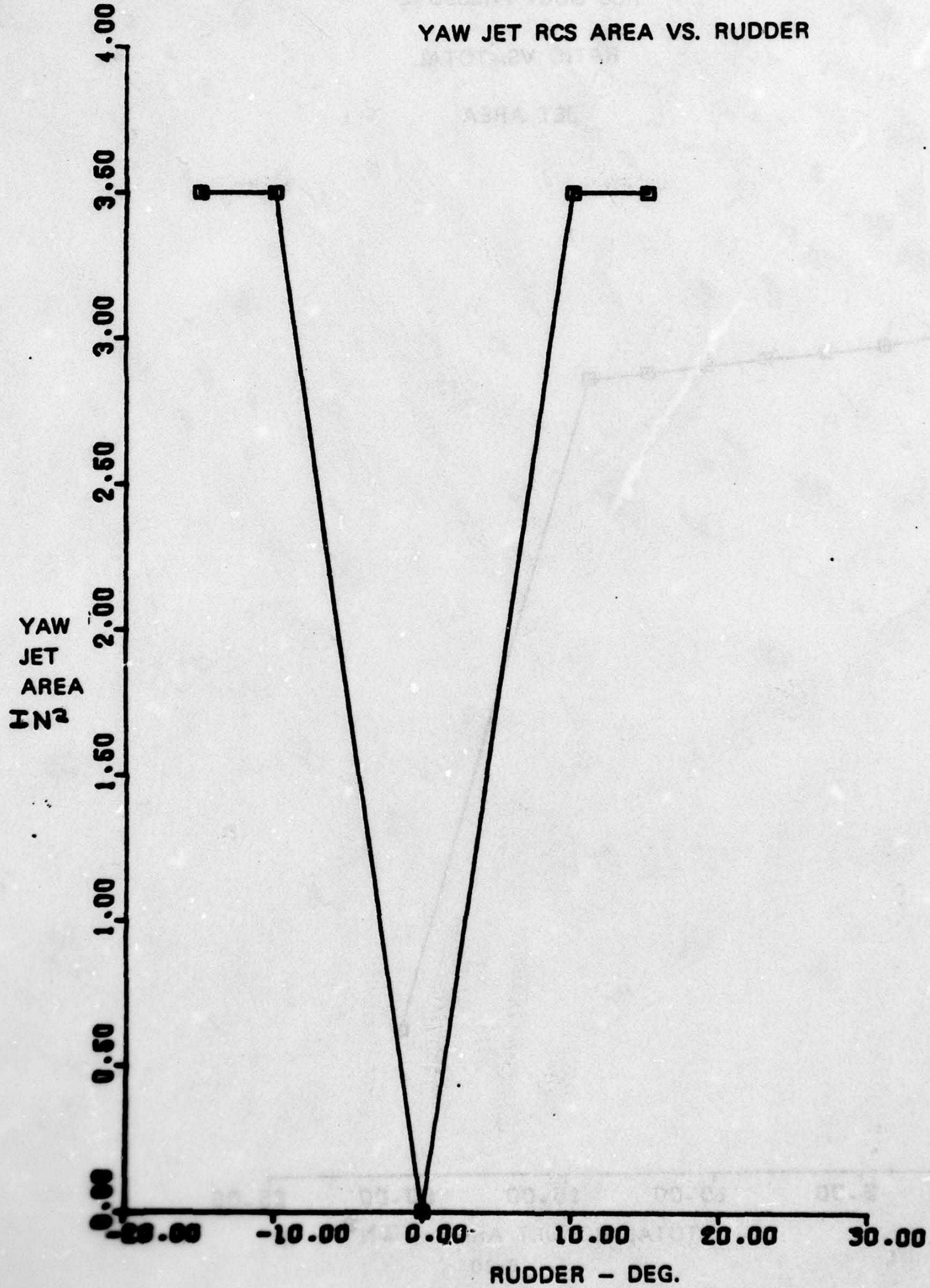

YAW JET RCS AREA VS. RUDDER
YAW
JET
AREA
IN²
4.00
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
-20.00
-10.00
0.00
10.00
20.00
30.00
RUDDER – DEG.

TABLE NO.8

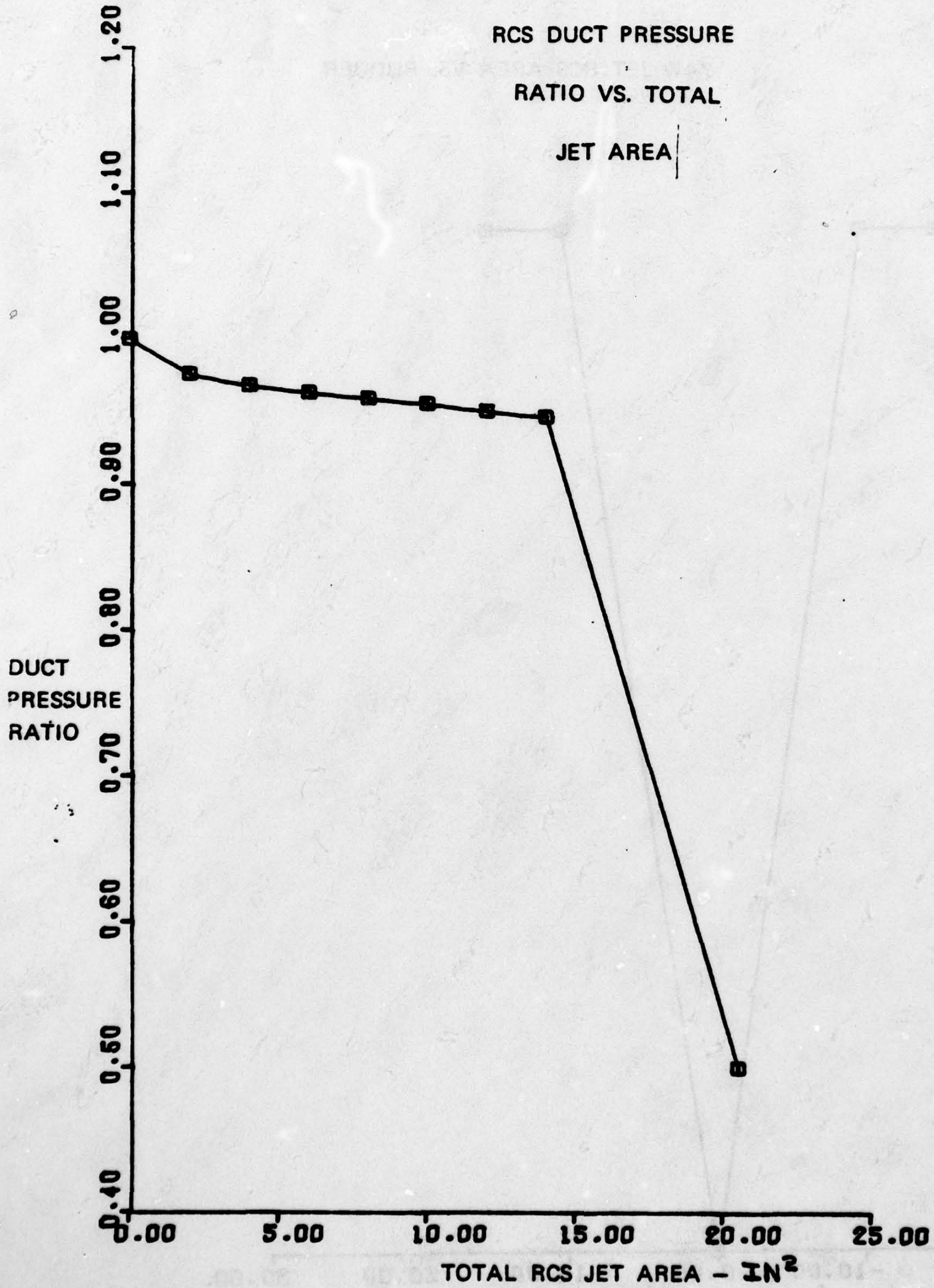

RCS DUCT PRESSURE
RATIO VS. TOTAL
JET AREA
DUCT
PRESSURE
RATIO
1.20
1.10
1.00
0.90
0.80
0.70
0.60
0.50
0.40
0.00
5.00
10.00
15.00
20.00
25.00
TOTAL RCS JET AREA - IN^2

TABLE NO.9

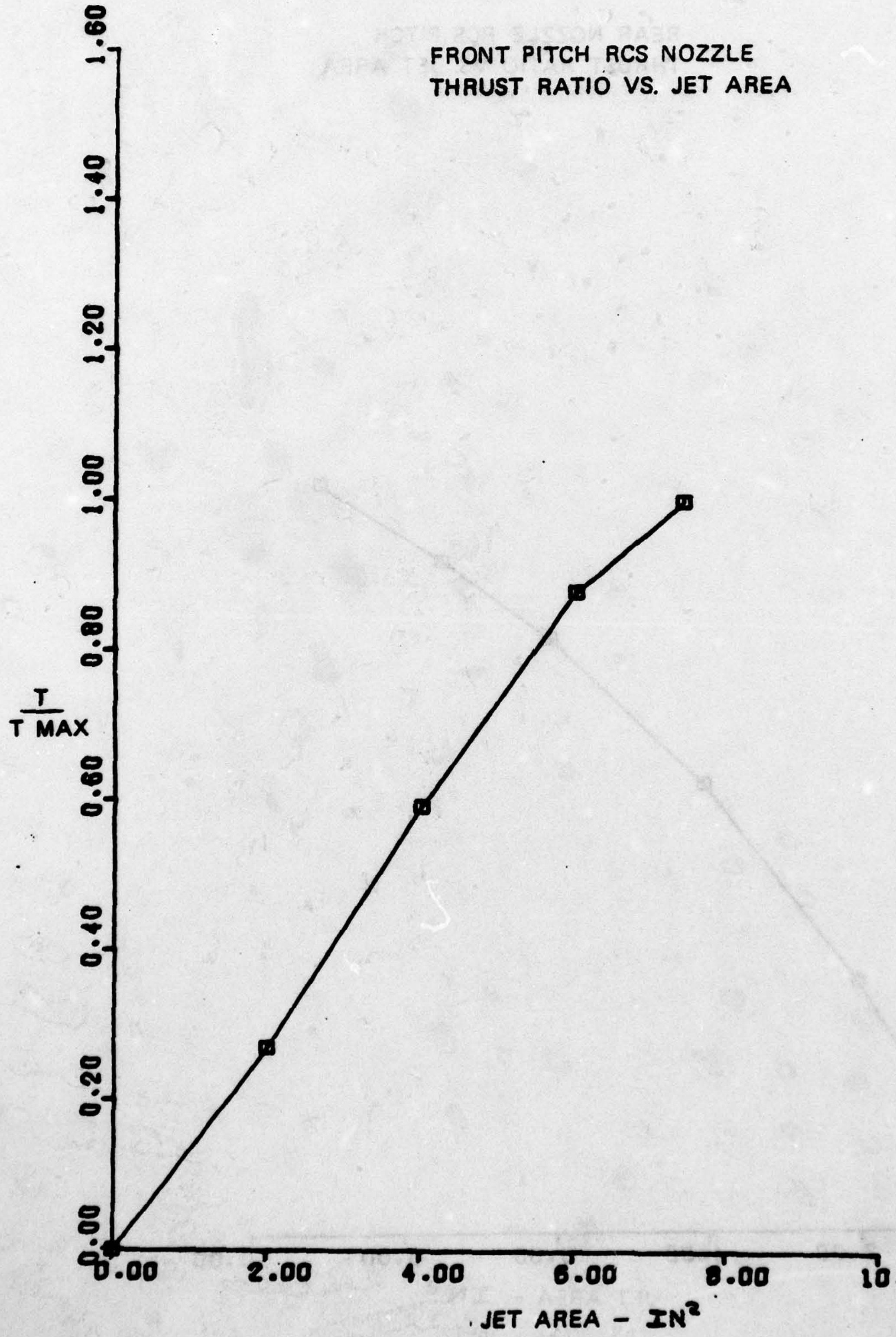

FRONT PITCH RCS NOZZLE
THRUST RATIO VS. JET AREA
1.60
1.40
1.20
1.00
0.80
0.60
0.40
0.20
0.00
T / T MAX
0.00
2.00
4.00
6.00
8.00
10.00
JET AREA – IN²

TABLE NO.10

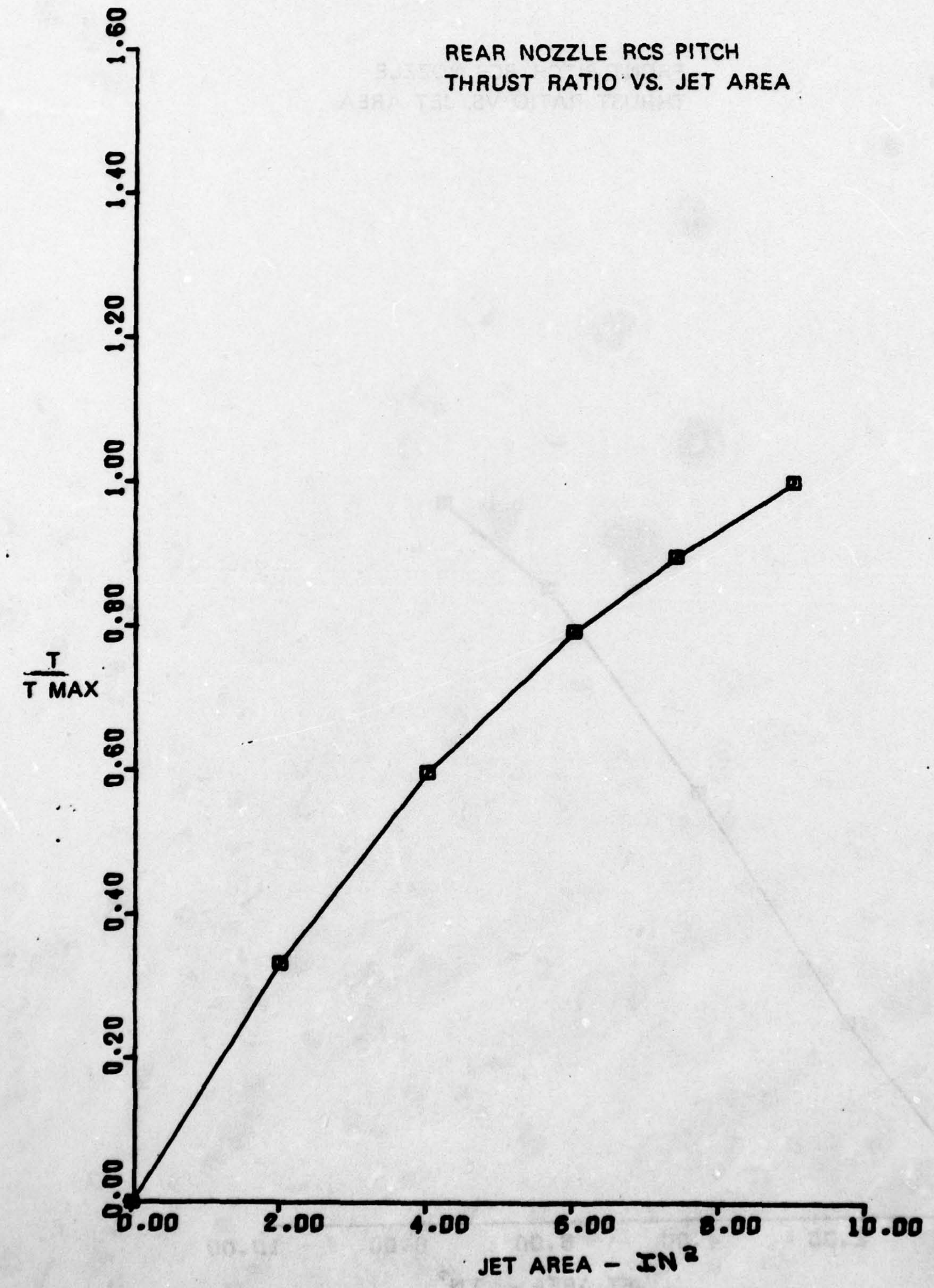

REAR NOZZLE RCS PITCH
THRUST RATIO VS. JET AREA
1.60
1.40
1.20
1.00
0.80
0.60
0.40
0.20
0.00
T / T MAX
0.00
2.00
4.00
6.00
8.00
10.00
JET AREA - IN²

TABLE NO.11

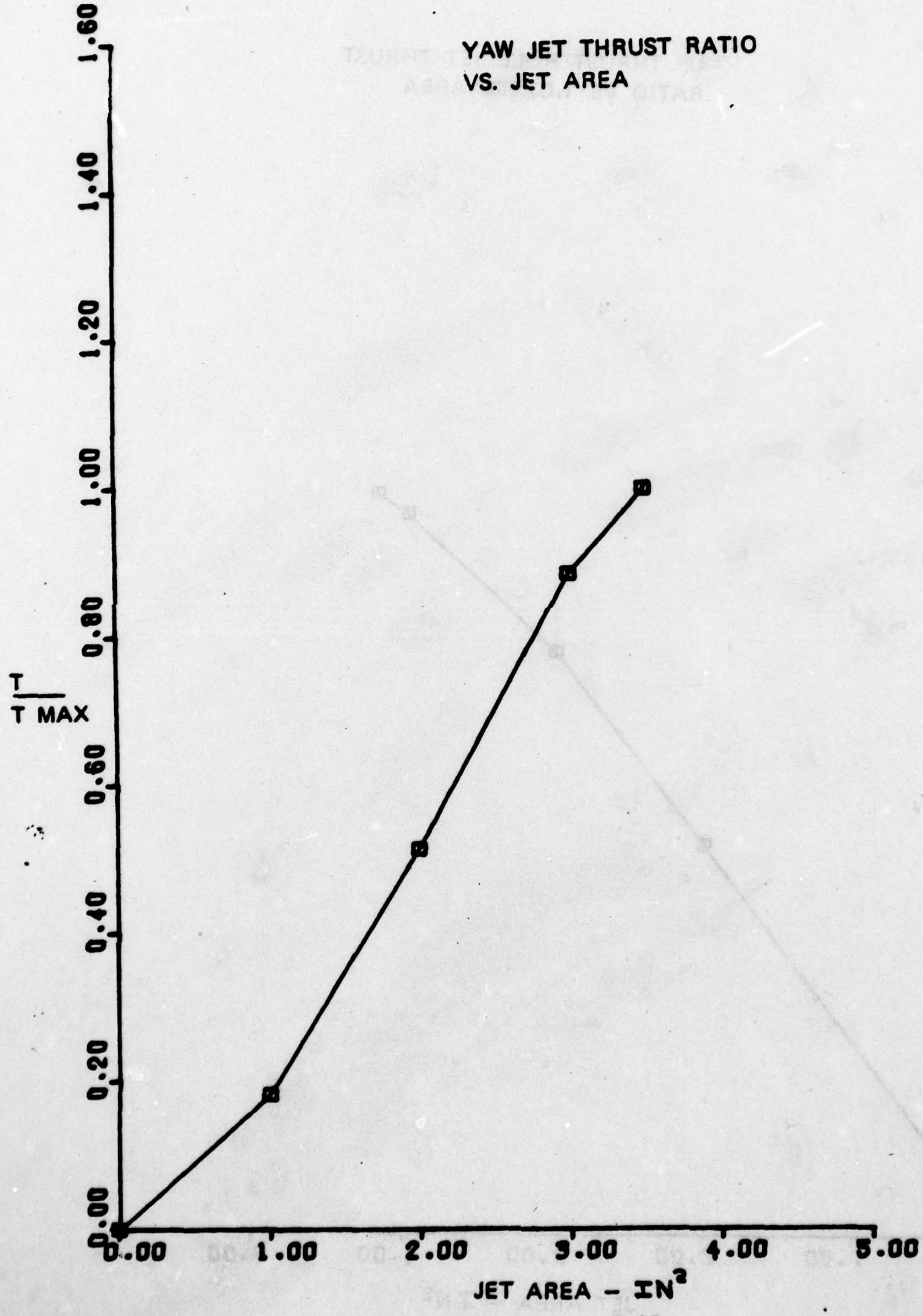

YAW JET THRUST RATIO
VS. JET AREA
T / T MAX
0.00
0.20
0.40
0.60
0.80
1.00
1.20
1.40
1.60
0.00
1.00
2.00
3.00
4.00
5.00
JET AREA - IN²

TABLE NO.12

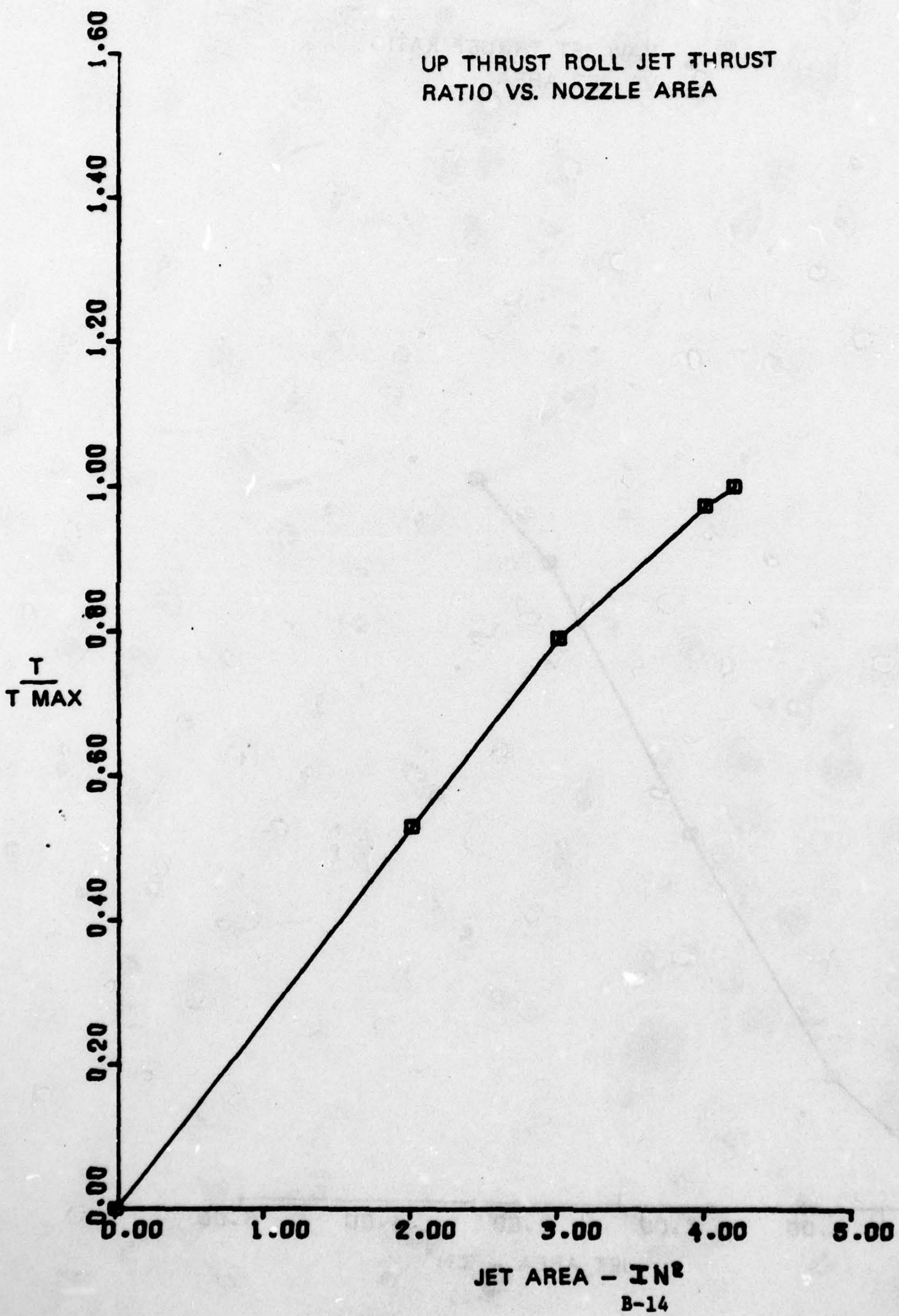

UP THRUST ROLL JET THRUST
RATIO VS. NOZZLE AREA
T / T MAX
0.00
0.20
0.40
0.60
0.80
1.00
1.20
1.40
1.60
0.00
1.00
2.00
3.00
4.00
5.00
JET AREA - IN²

TABLE NO.13

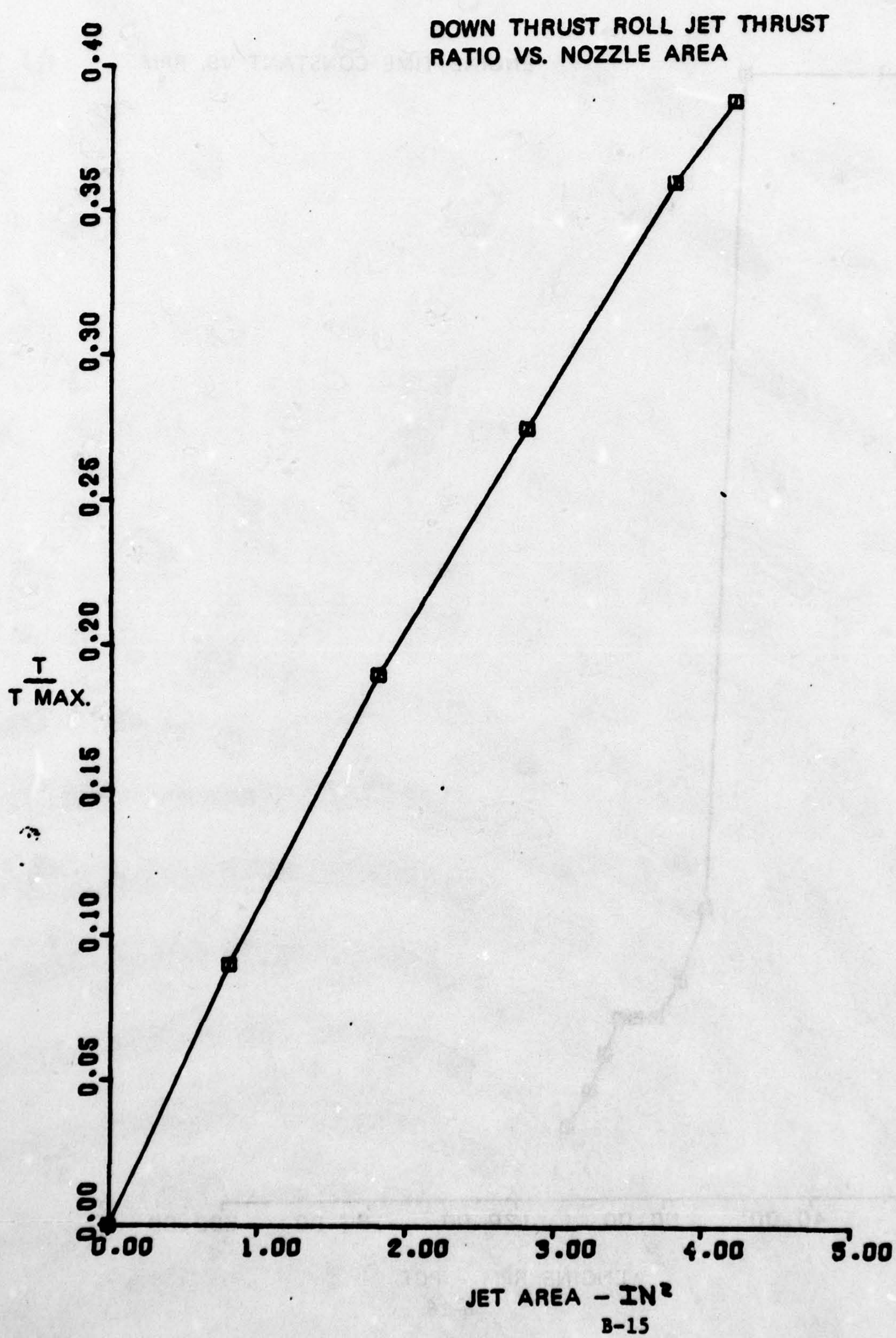

DOWN THRUST ROLL JET THRUST
RATIO VS. NOZZLE AREA
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
T / T MAX.
0.00
1.00
2.00
3.00
4.00
5.00
JET AREA – IN²

TABLE NO.14

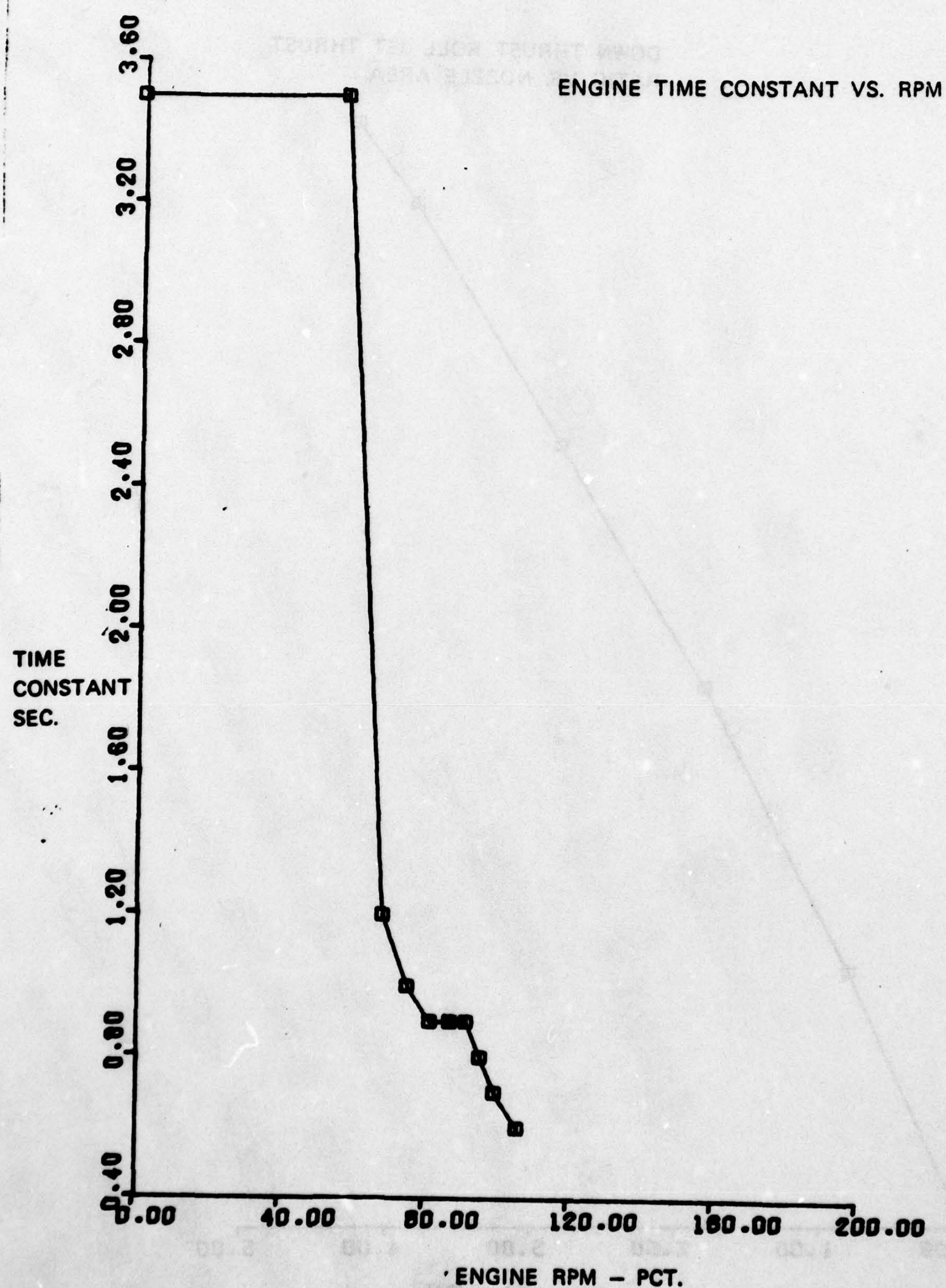

ENGINE TIME CONSTANT VS. RPM
TIME
CONSTANT
SEC.
3.60
3.20
2.80
2.40
2.00
1.60
1.20
0.80
0.40
0.00
40.00
80.00
120.00
160.00
200.00
ENGINE RPM - PCT.

TABLE NO.15

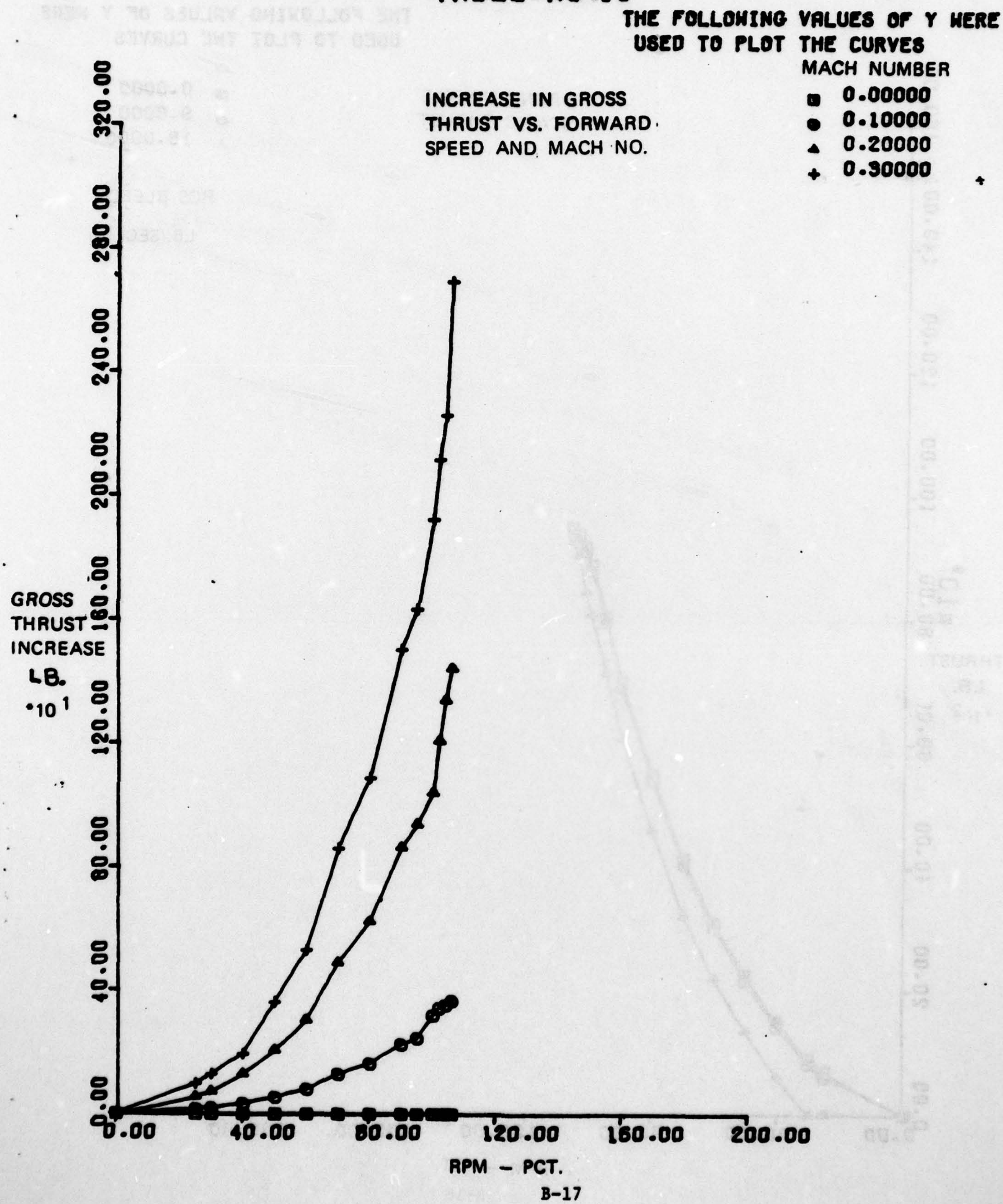

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
MACH NUMBER
0.00000
0.10000
0.20000
0.30000
INCREASE IN GROSS THRUST VS. FORWARD SPEED AND MACH NO.
GROSS THRUST INCREASE LB. *10 1
0.00
40.00
80.00
120.00
160.00
200.00
240.00
280.00
320.00
0.00
40.00
80.00
120.00
160.00
200.00
RPM - PCT.

TABLE NO.16

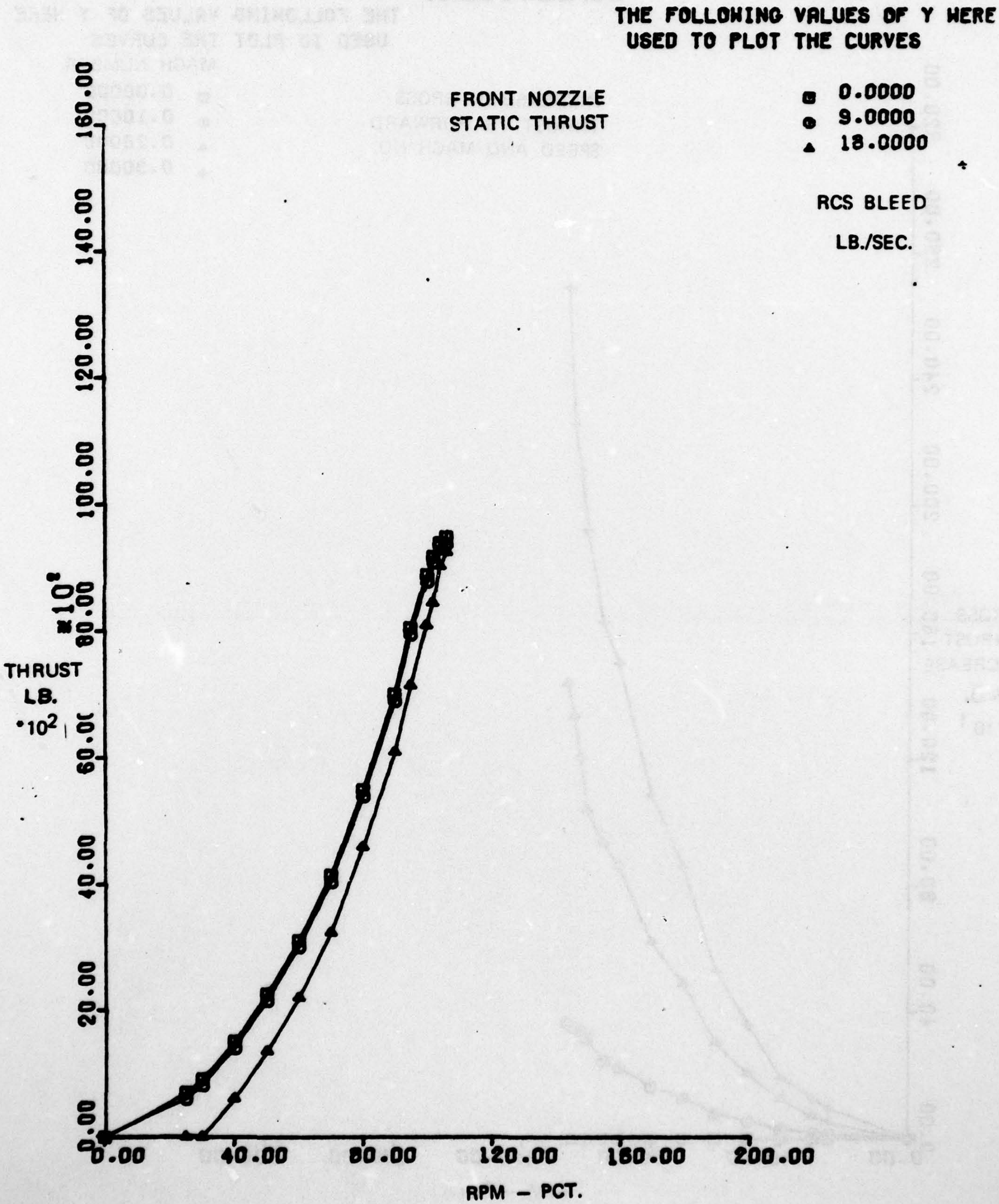

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
FRONT NOZZLE
STATIC THRUST
0.0000
9.0000
18.0000
RCS BLEED
LB./SEC.
THRUST
LB.
$\cdot 10^2$
0.00
20.00
40.00
60.00
80.00
100.00
120.00
140.00
160.00
0.00
40.00
80.00
120.00
160.00
200.00
RPM — PCT.

TABLE NO.17

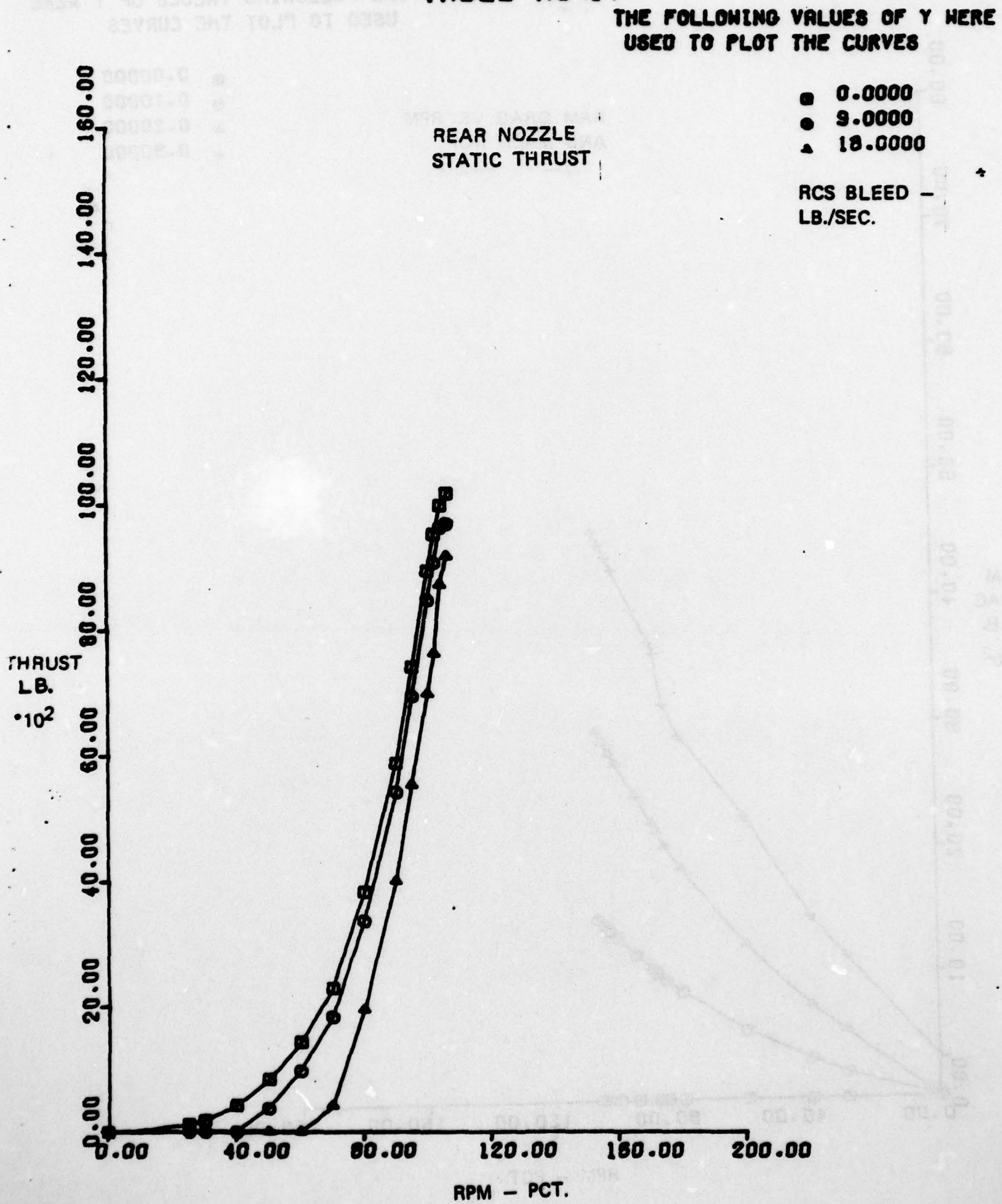

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
0.0000
9.0000
18.0000
RCS BLEED — LB./SEC.
REAR NOZZLE
STATIC THRUST
THRUST
LB.
*10²
0.00
20.00
40.00
60.00
80.00
100.00
120.00
140.00
160.00
0.00
40.00
80.00
120.00
160.00
200.00
RPM — PCT.

TABLE NO.18

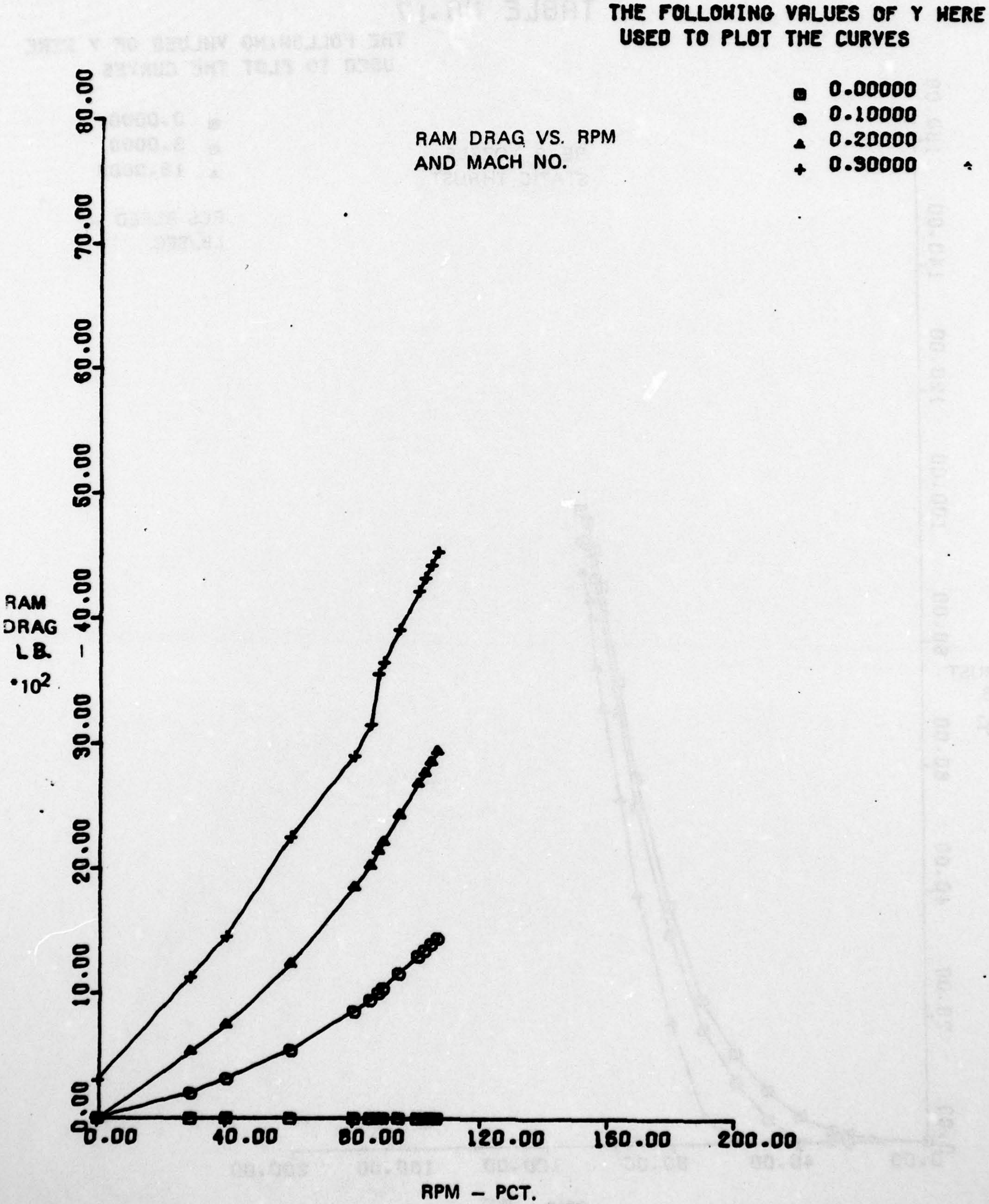

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
0.00000
0.10000
0.20000
0.30000
RAM DRAG VS. RPM AND MACH NO.
RAM DRAG LB. - *10^2
RPM - PCT.
0.00
10.00
20.00
30.00
40.00
50.00
60.00
70.00
80.00
0.00
40.00
80.00
120.00
160.00
200.00

TABLE NO.19

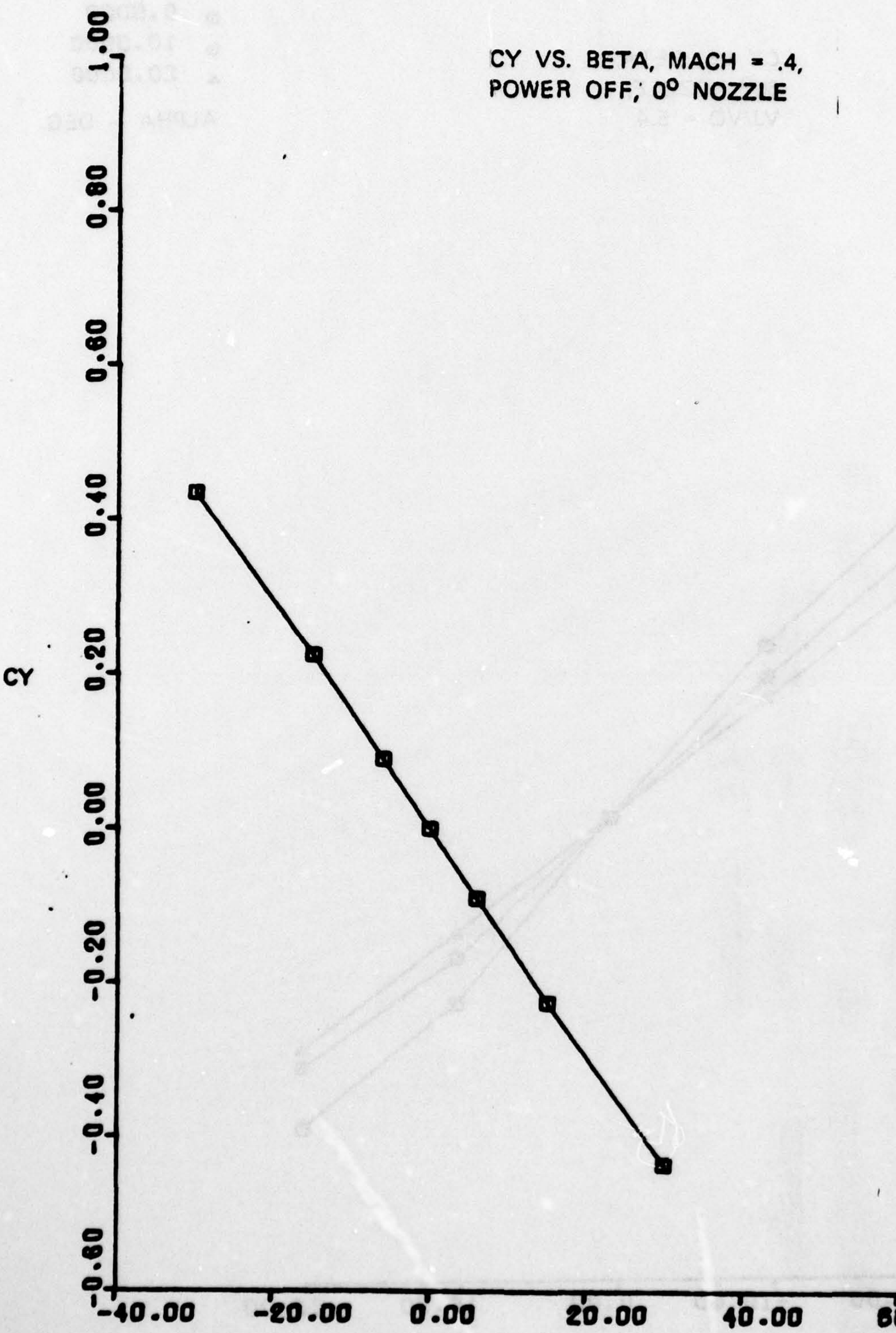
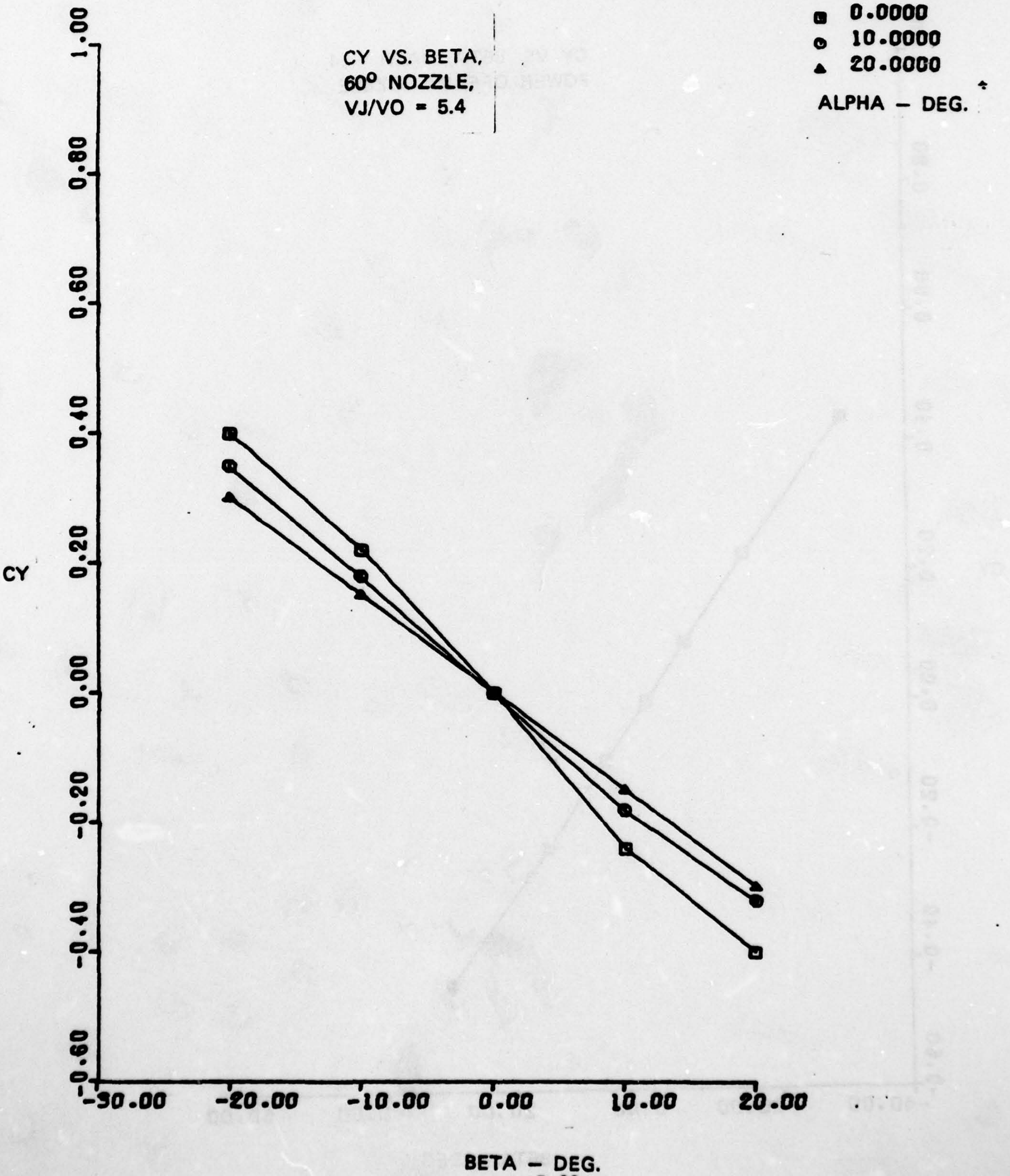

CY VS. BETA, MACH = .4,
POWER OFF, 0° NOZZLE
CY
1.00
0.80
0.60
0.40
0.20
0.00
-0.20
-0.40
-0.60
-40.00
-20.00
0.00
20.00
40.00
60.00
BETA – DEG.

TABLE NO 20

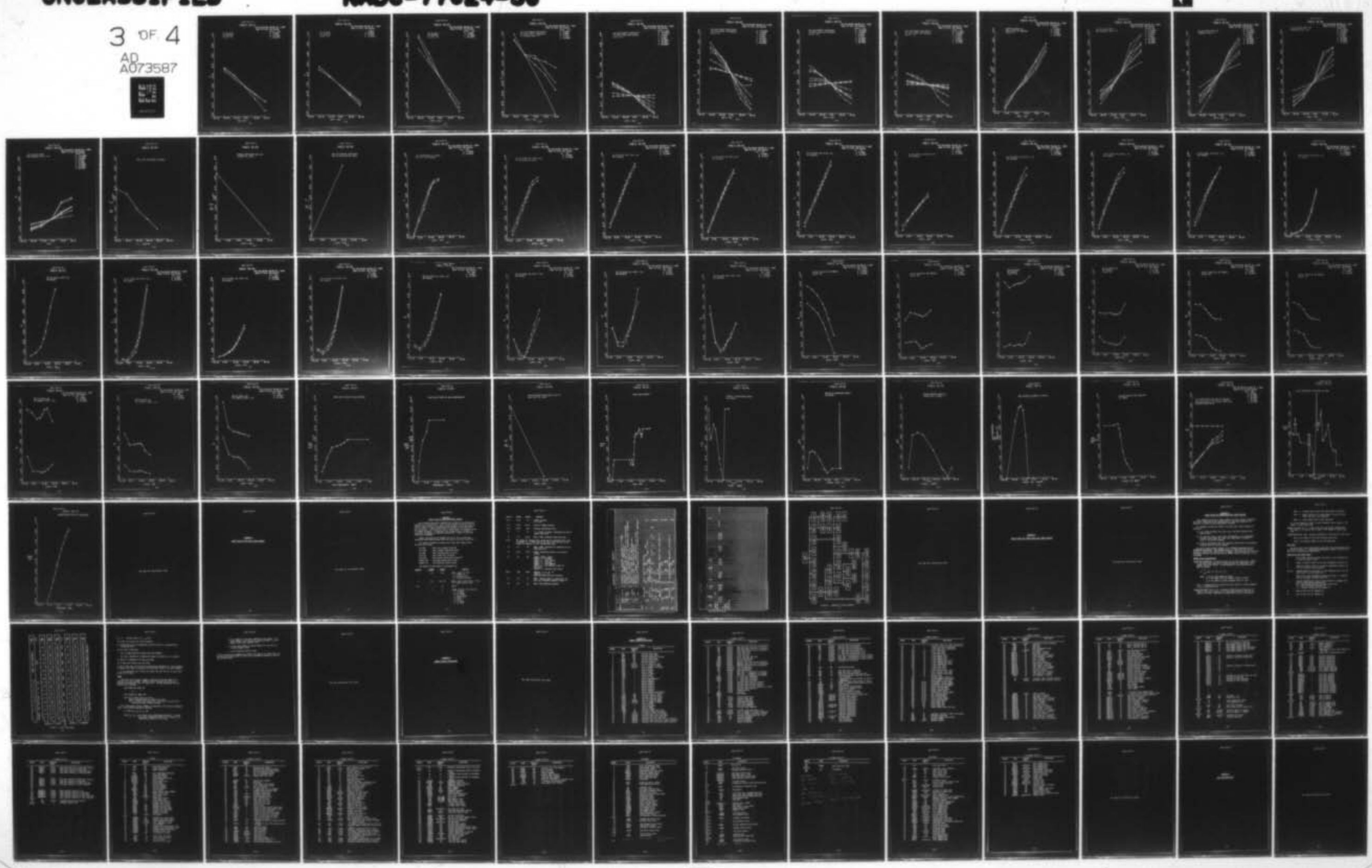
THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES

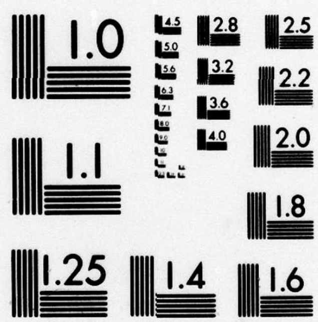


AD-A073 587 NAVAL AIR DEVELOPMENT CENTER WARMINSTER PA AIR VEHICL--ETC F/G 1/2
A COMPUTERIZED VSTOL/SMALL PLATFORM LANDING DYNAMICS INVESTIGAT--ETC(U)
SEP 77 R L NAVE

UNCLASSIFIED NADC-77024-30 NL

3 OF 4
AD
A073587





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

TABLE NO.21

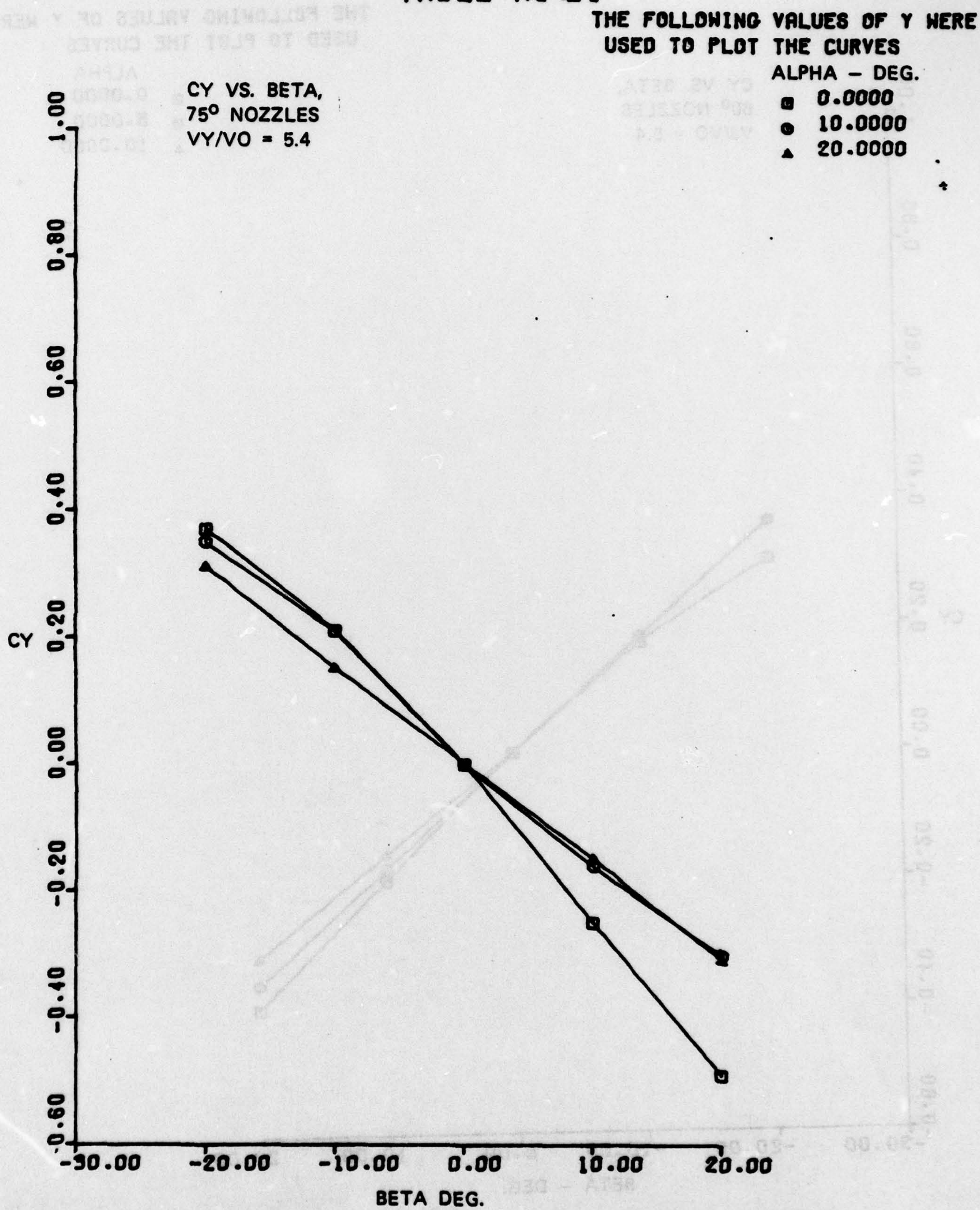

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
ALPHA - DEG.
0.0000
10.0000
20.0000
CY VS. BETA,
75° NOZZLES
VY/VO = 5.4
CY
1.00
0.80
0.60
0.40
0.20
0.00
-0.20
-0.40
-0.60
-30.00
-20.00
-10.00
0.00
10.00
20.00
BETA DEG.

TABLE NO 22

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES

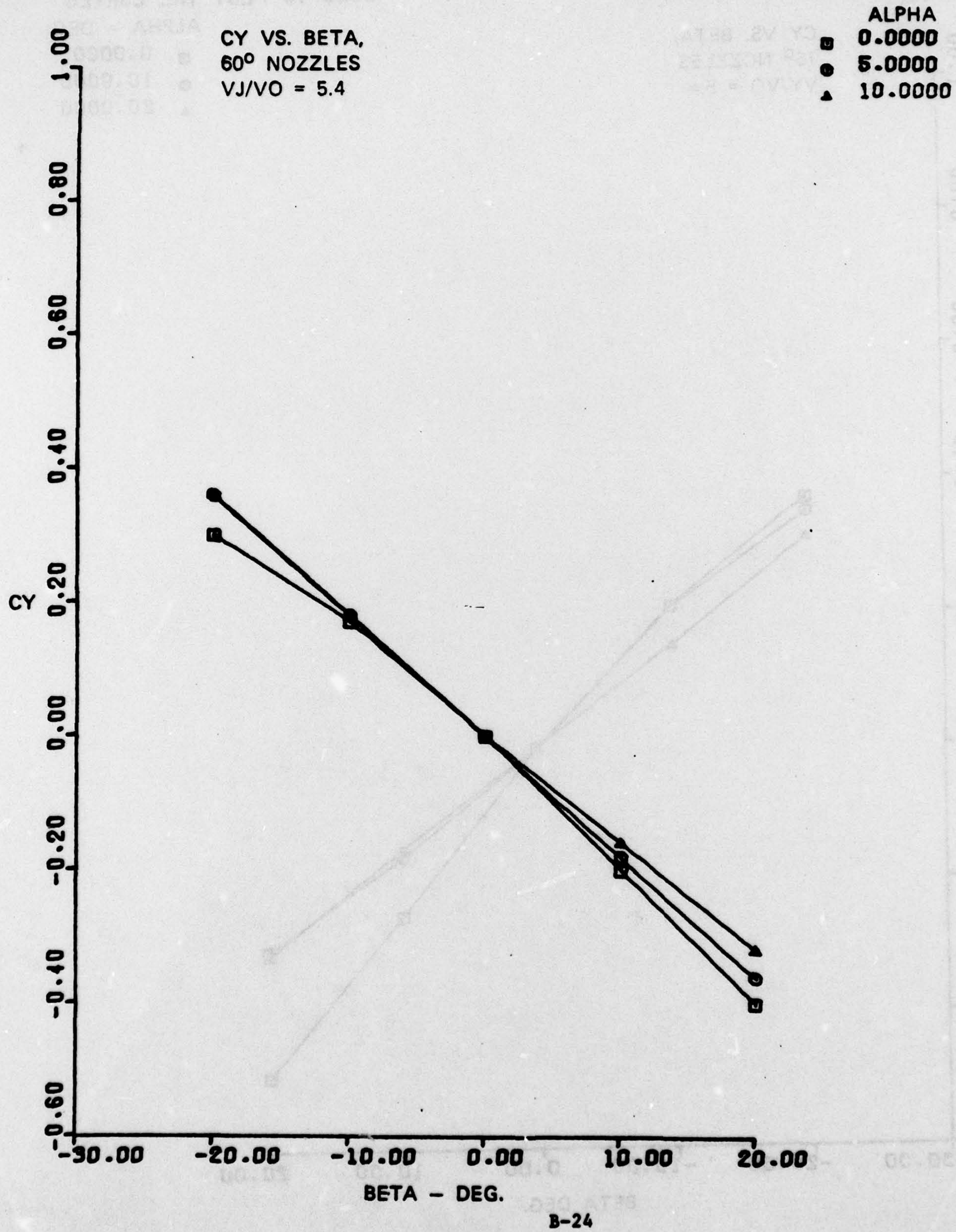


TABLE NO 23

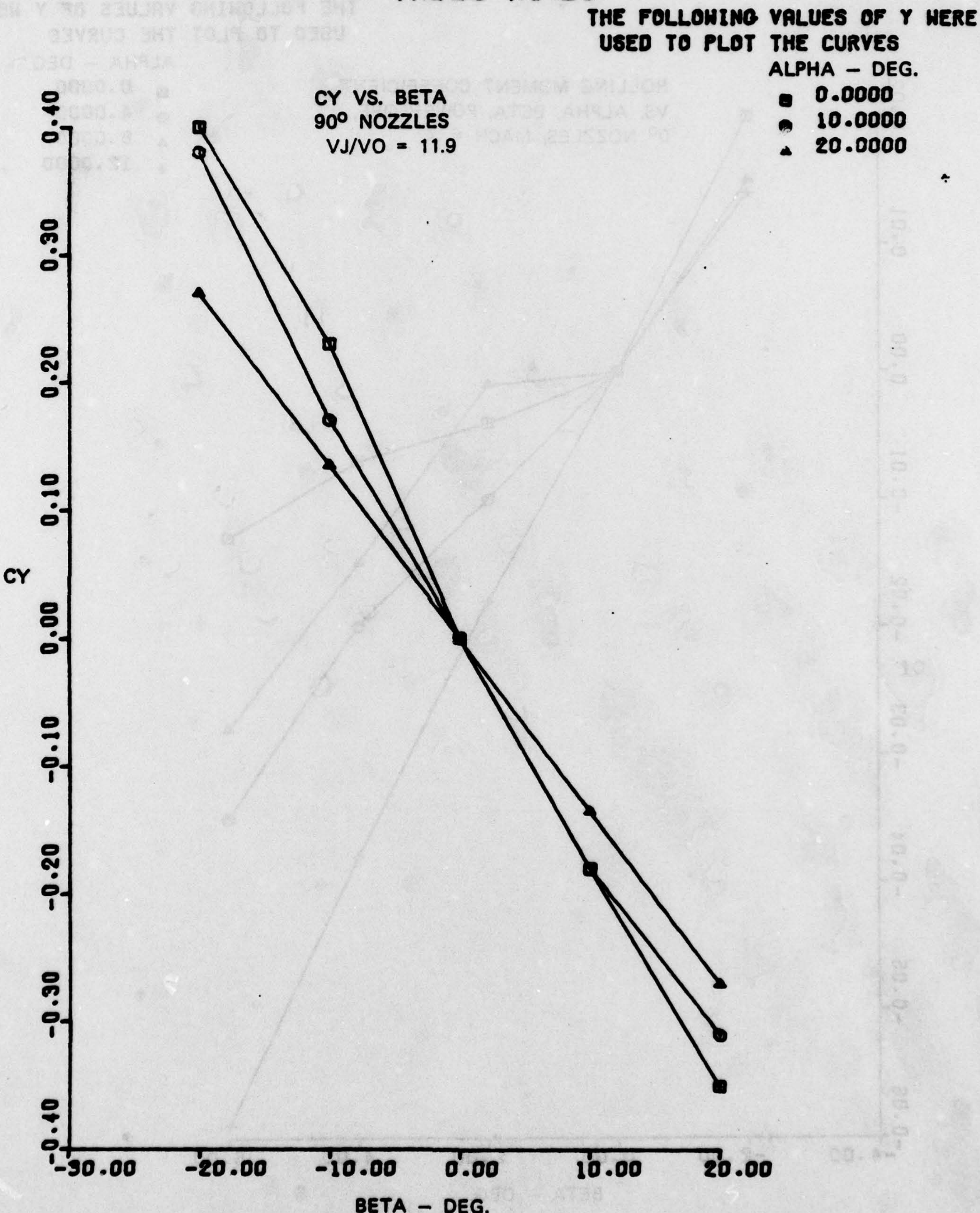

THE FOLLOWING VALUES OF Y WERE
USED TO PLOT THE CURVES
ALPHA – DEG.
0.0000
10.0000
20.0000
CY VS. BETA
90° NOZZLES
VJ/VO = 11.9
CY
0.40
0.30
0.20
0.10
0.00
-0.10
-0.20
-0.30
-0.40
-30.00
-20.00
-10.00
0.00
10.00
20.00
BETA – DEG.

TABLE NO.24

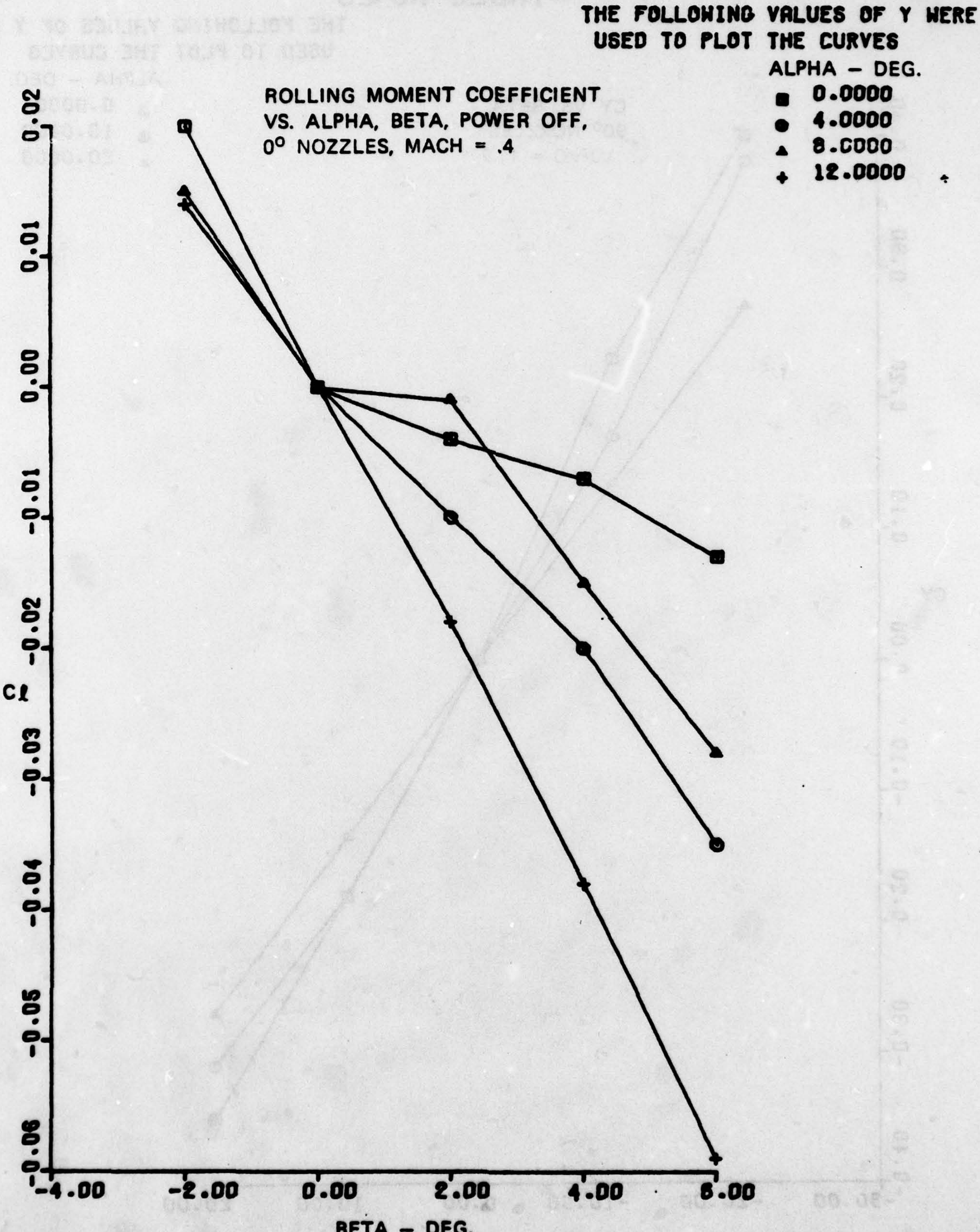

THE FOLLOWING VALUES OF Y WERE
USED TO PLOT THE CURVES
ALPHA – DEG.
0.0000
4.0000
8.0000
12.0000
ROLLING MOMENT COEFFICIENT
VS. ALPHA, BETA, POWER OFF,
0° NOZZLES, MACH = .4
Cℓ
0.02
0.01
0.00
-0.01
-0.02
-0.03
-0.04
-0.05
-0.06
-4.00
-2.00
0.00
2.00
4.00
6.00
BETA – DEG.

TABLE NO.25

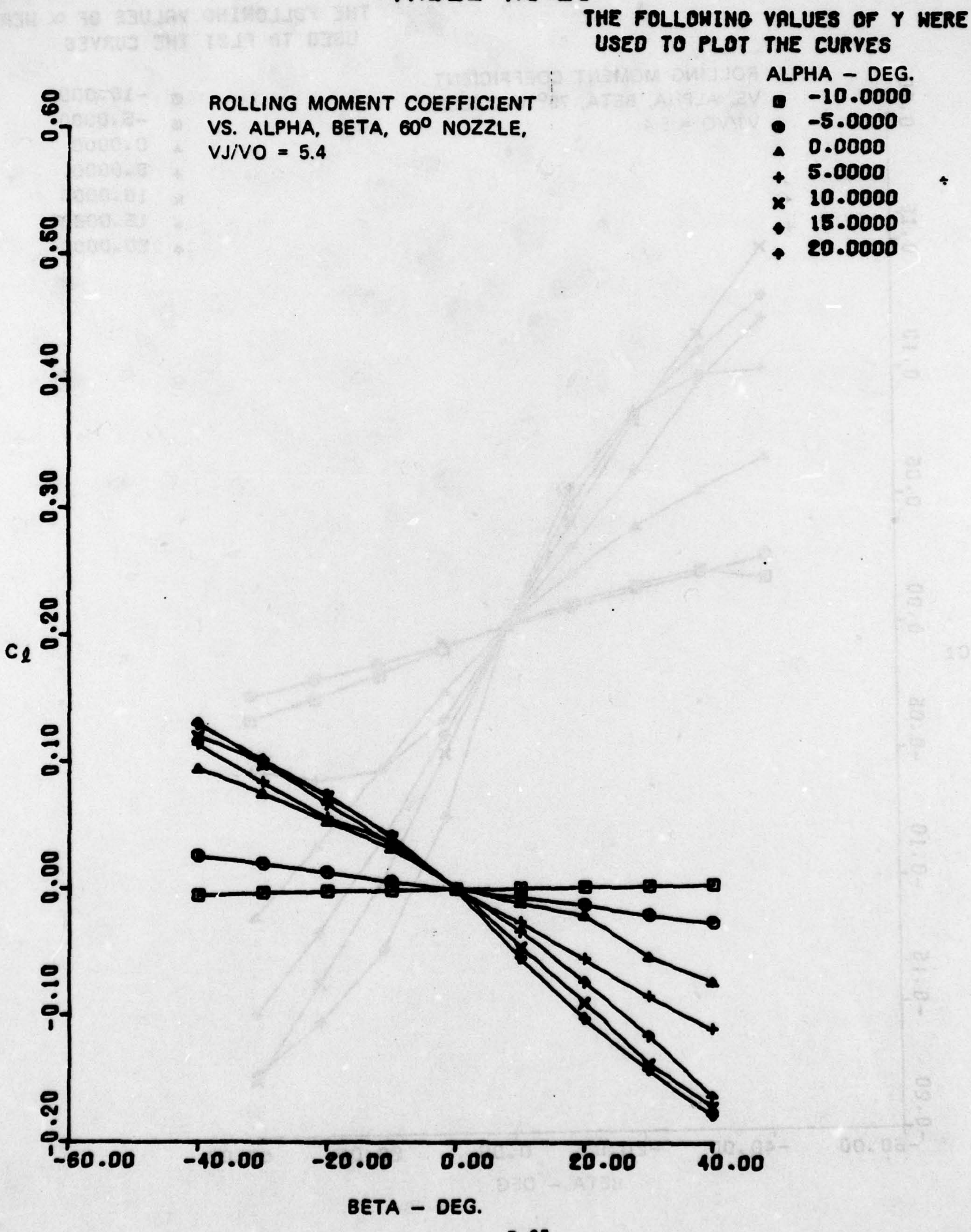

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
ALPHA - DEG.
-10.0000
-5.0000
0.0000
5.0000
10.0000
15.0000
20.0000
ROLLING MOMENT COEFFICIENT
VS. ALPHA, BETA, 60° NOZZLE,
VJ/VO = 5.4
Cl
0.60
0.50
0.40
0.30
0.20
0.10
0.00
-0.10
-0.20
-60.00
-40.00
-20.00
0.00
20.00
40.00
BETA - DEG.

TABLE NO 26

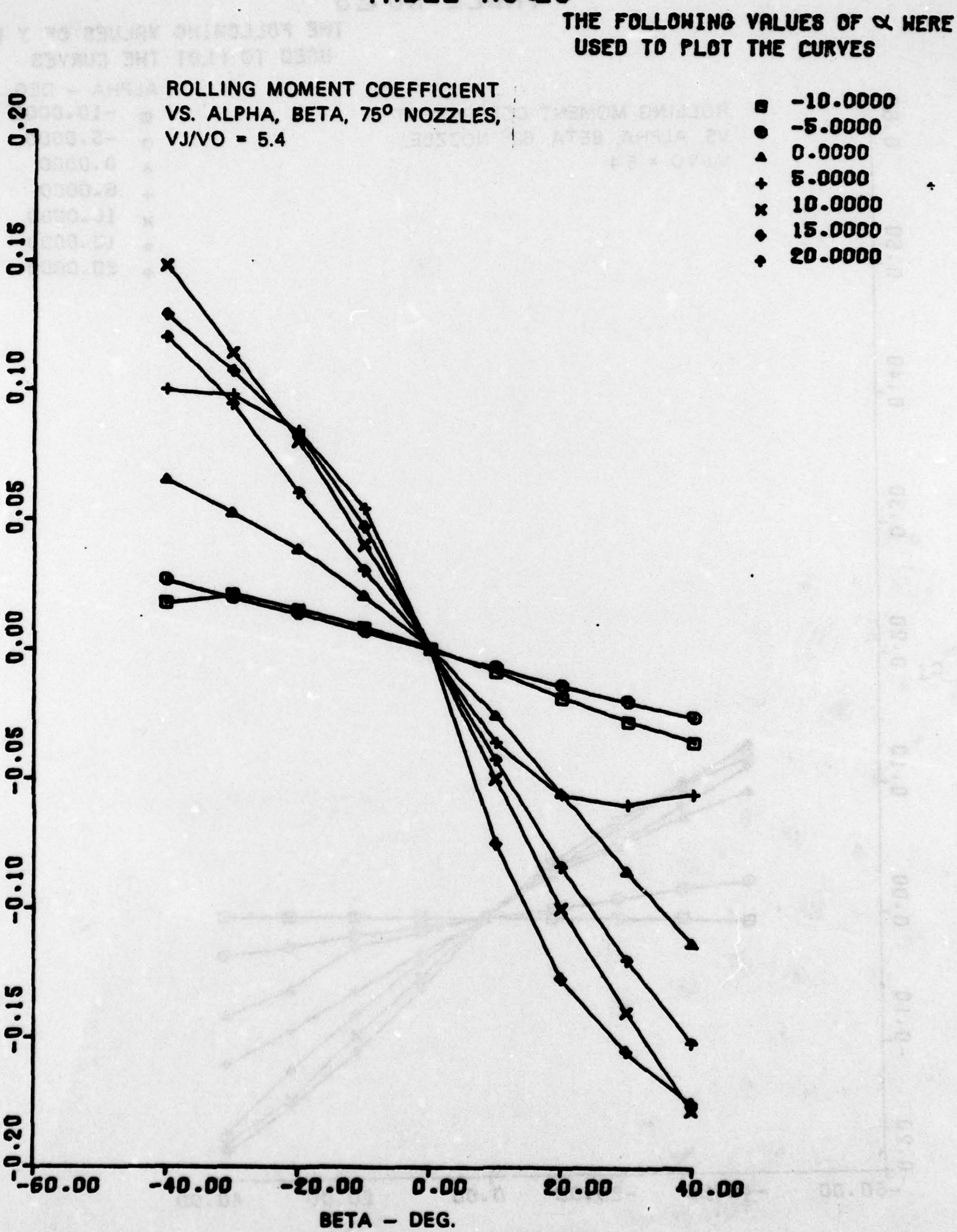

THE FOLLOWING VALUES OF α WERE USED TO PLOT THE CURVES
ROLLING MOMENT COEFFICIENT VS. ALPHA, BETA, 75° NOZZLES, VJ/VO = 5.4
-10.0000
-5.0000
0.0000
5.0000
10.0000
15.0000
20.0000
0.20
0.15
0.10
0.05
0.00
-0.05
-0.10
-0.15
-0.20
Cl
-60.00
-40.00
-20.00
0.00
20.00
40.00
BETA - DEG.

TABLE NO.27

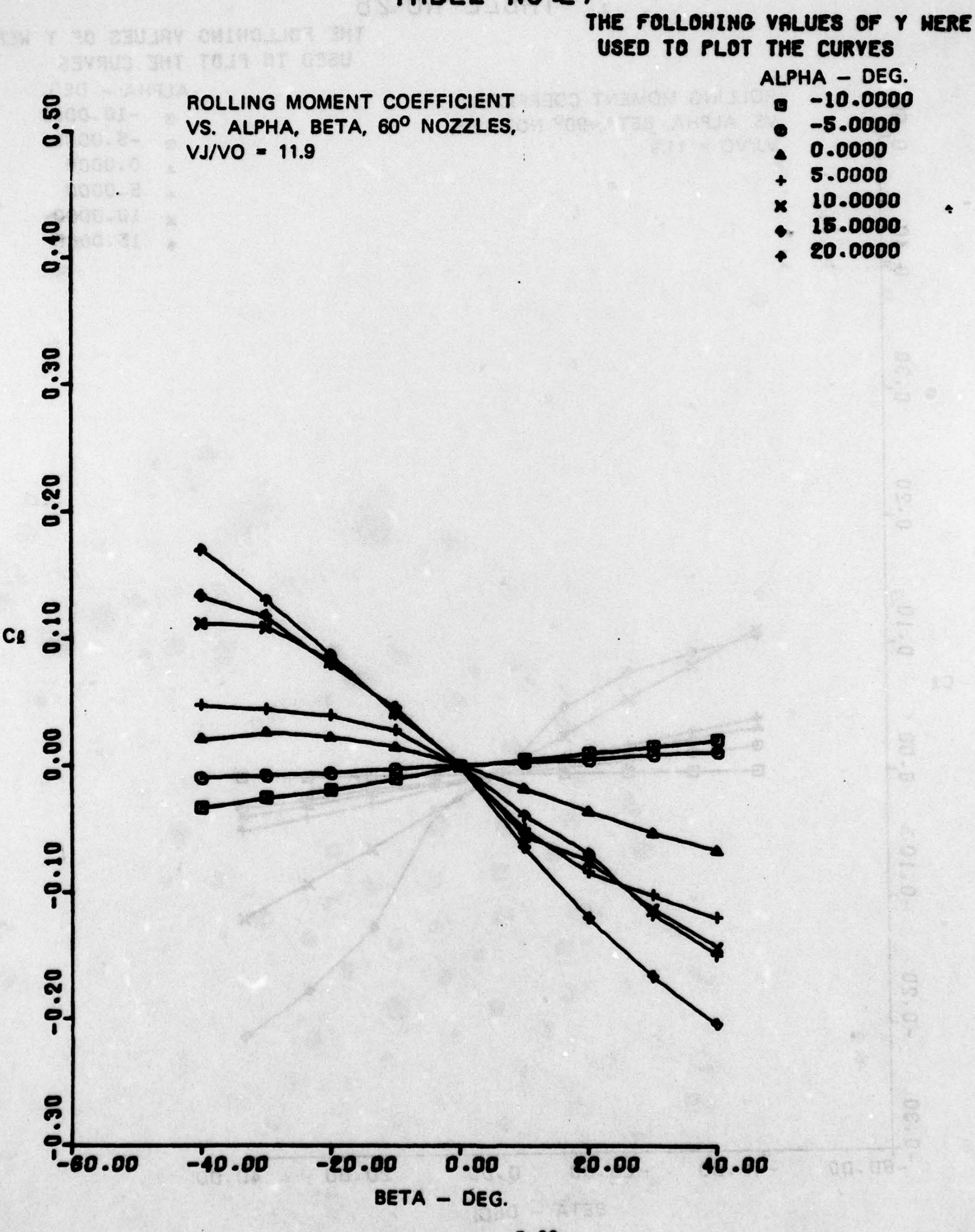

ROLLING MOMENT COEFFICIENT
VS. ALPHA, BETA, 60° NOZZLES,
VJ/VO = 11.9
THE FOLLOWING VALUES OF Y WERE
USED TO PLOT THE CURVES
ALPHA - DEG.
-10.0000
-5.0000
0.0000
5.0000
10.0000
15.0000
20.0000
Cℓ
0.50
0.40
0.30
0.20
0.10
0.00
-0.10
-0.20
-0.30
-60.00
-40.00
-20.00
0.00
20.00
40.00
BETA - DEG.

TABLE NO.28

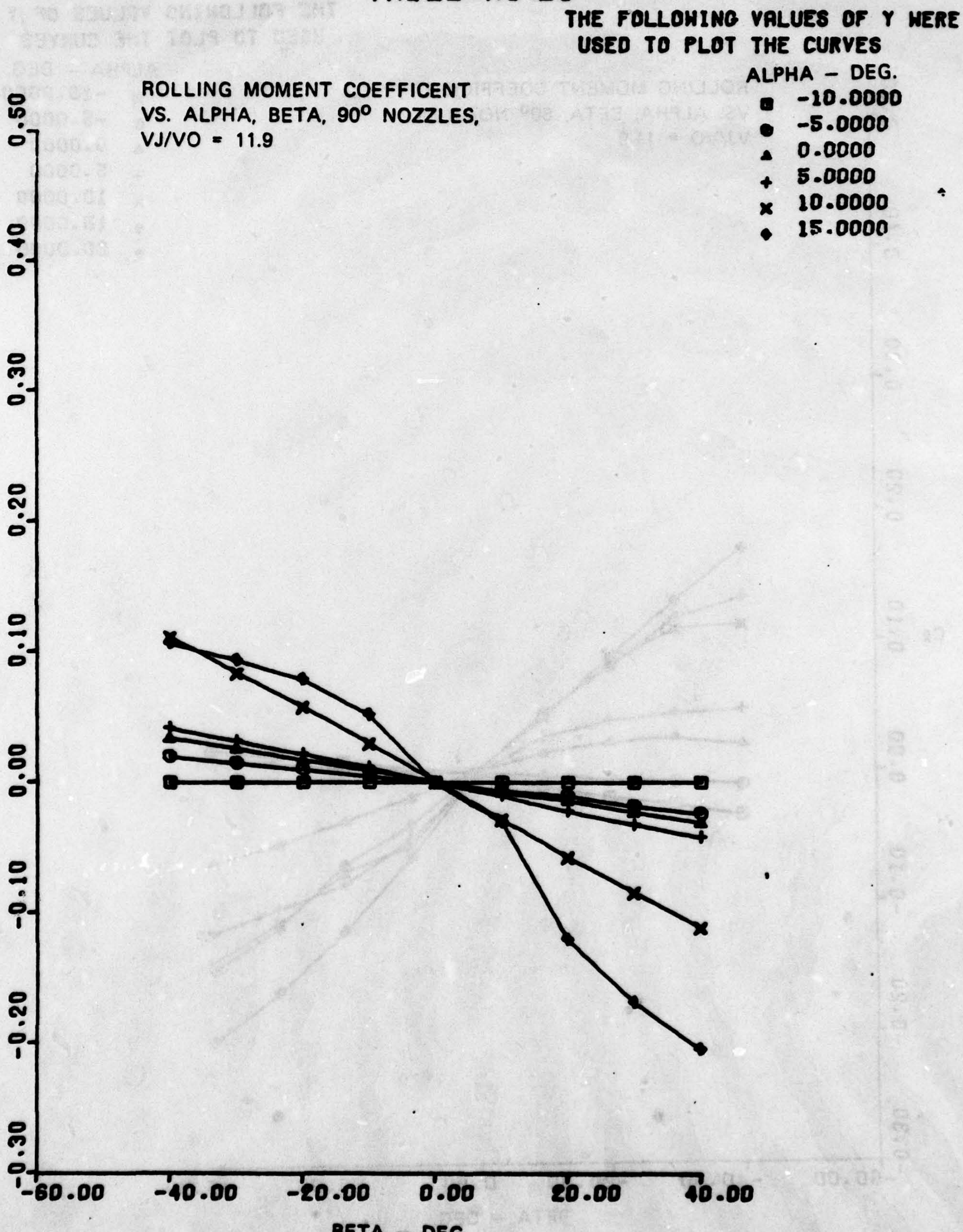

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
ALPHA – DEG.
-10.0000
-5.0000
0.0000
5.0000
10.0000
15.0000
ROLLING MOMENT COEFFICENT
VS. ALPHA, BETA, 90° NOZZLES,
VJ/VO = 11.9
0.50
0.40
0.30
0.20
0.10
0.00
-0.10
-0.20
-0.30
Cℓ
-60.00
-40.00
-20.00
0.00
20.00
40.00
BETA – DEG.

TABLE NO.29

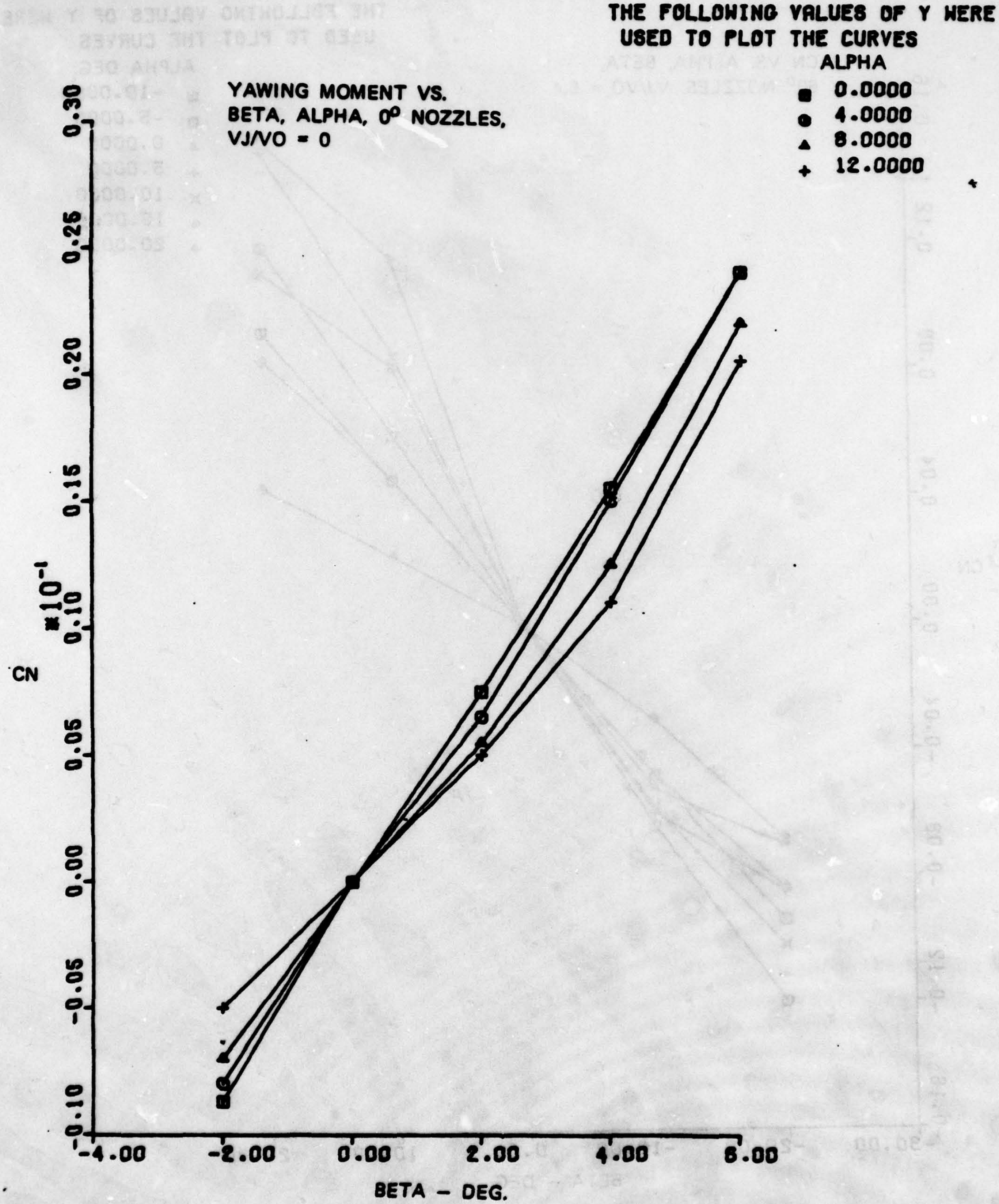

THE FOLLOWING VALUES OF Y WERE
USED TO PLOT THE CURVES
ALPHA
0.0000
4.0000
8.0000
12.0000
YAWING MOMENT VS.
BETA, ALPHA, 0° NOZZLES,
VJ/VO = 0
CN
$*10^{-1}$
0.30
0.25
0.20
0.15
0.10
0.05
0.00
-0.05
-0.10
-4.00
-2.00
0.00
2.00
4.00
6.00
BETA - DEG.

TABLE NO.30

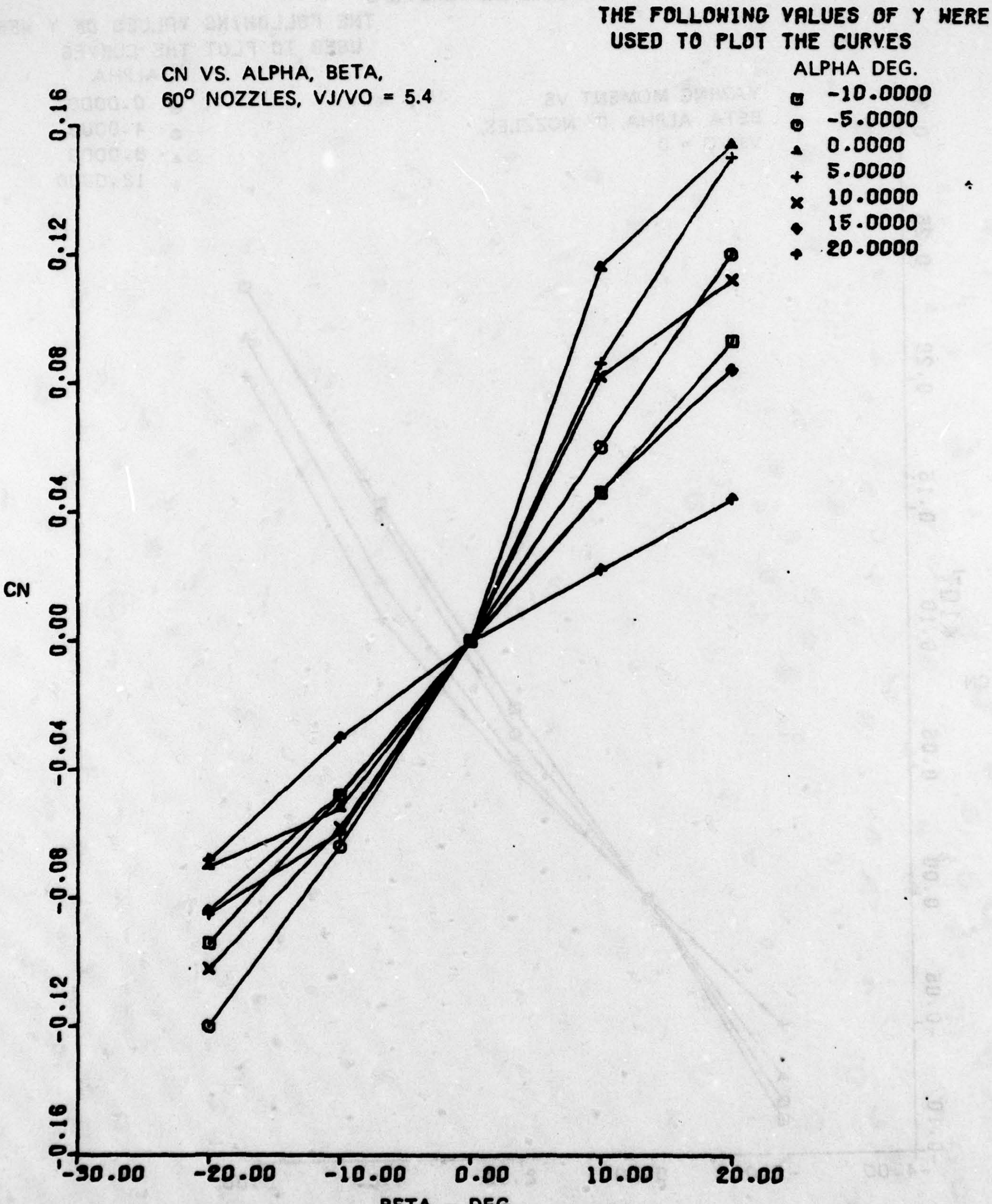

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
ALPHA DEG.
-10.0000
-5.0000
0.0000
5.0000
10.0000
15.0000
20.0000
CN VS. ALPHA, BETA, 60° NOZZLES, VJ/VO = 5.4
CN
0.16
0.12
0.08
0.04
0.00
-0.04
-0.08
-0.12
-0.16
-30.00
-20.00
-10.00
0.00
10.00
20.00
BETA - DEG.

TABLE NO.31

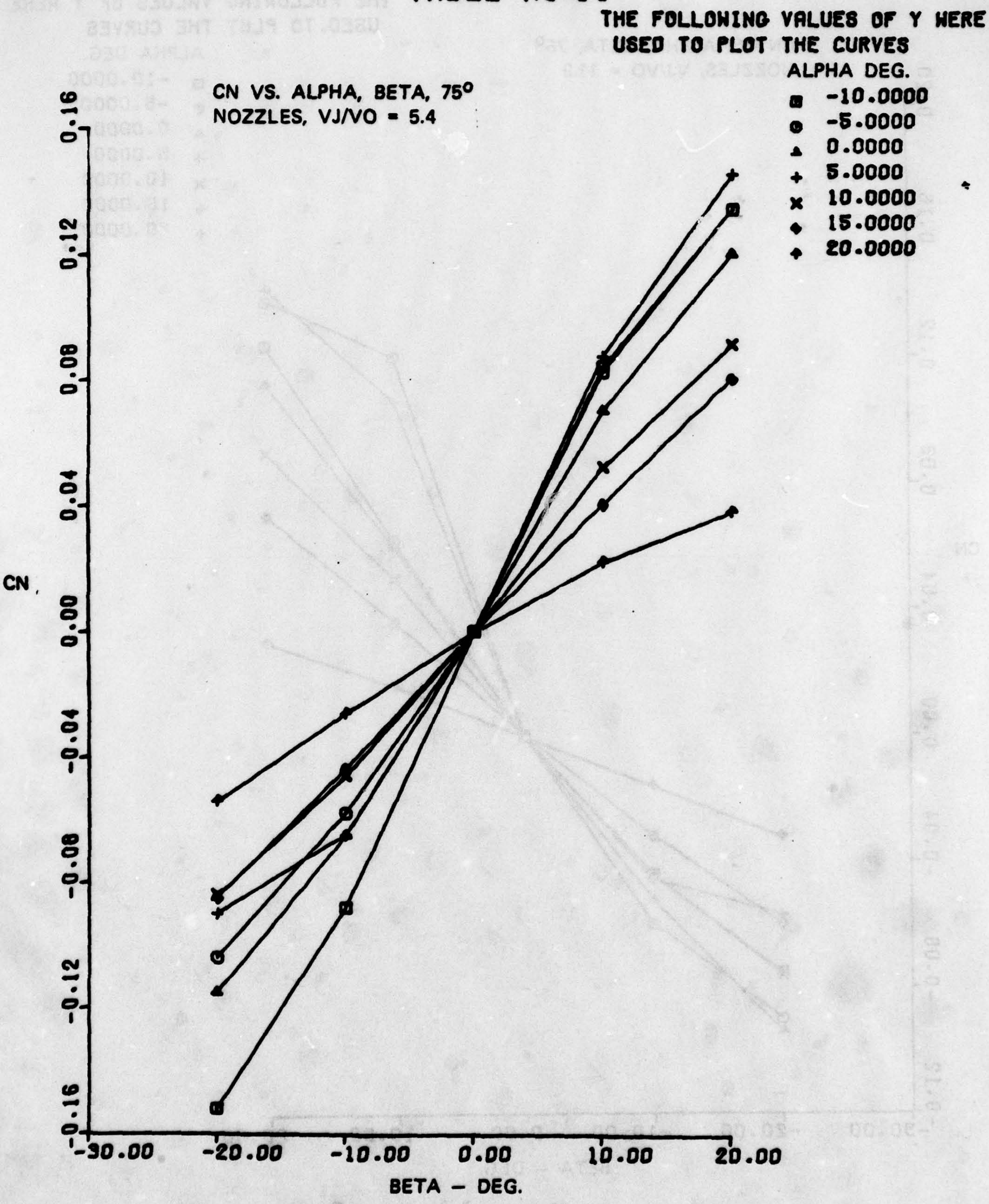

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
ALPHA DEG.
-10.0000
-5.0000
0.0000
5.0000
10.0000
15.0000
20.0000
CN VS. ALPHA, BETA, 75°
NOZZLES, VJ/VO = 5.4
CN
0.16
0.12
0.08
0.04
0.00
-0.04
-0.08
-0.12
-0.16
-30.00
-20.00
-10.00
0.00
10.00
20.00
BETA – DEG.

TABLE NO.32

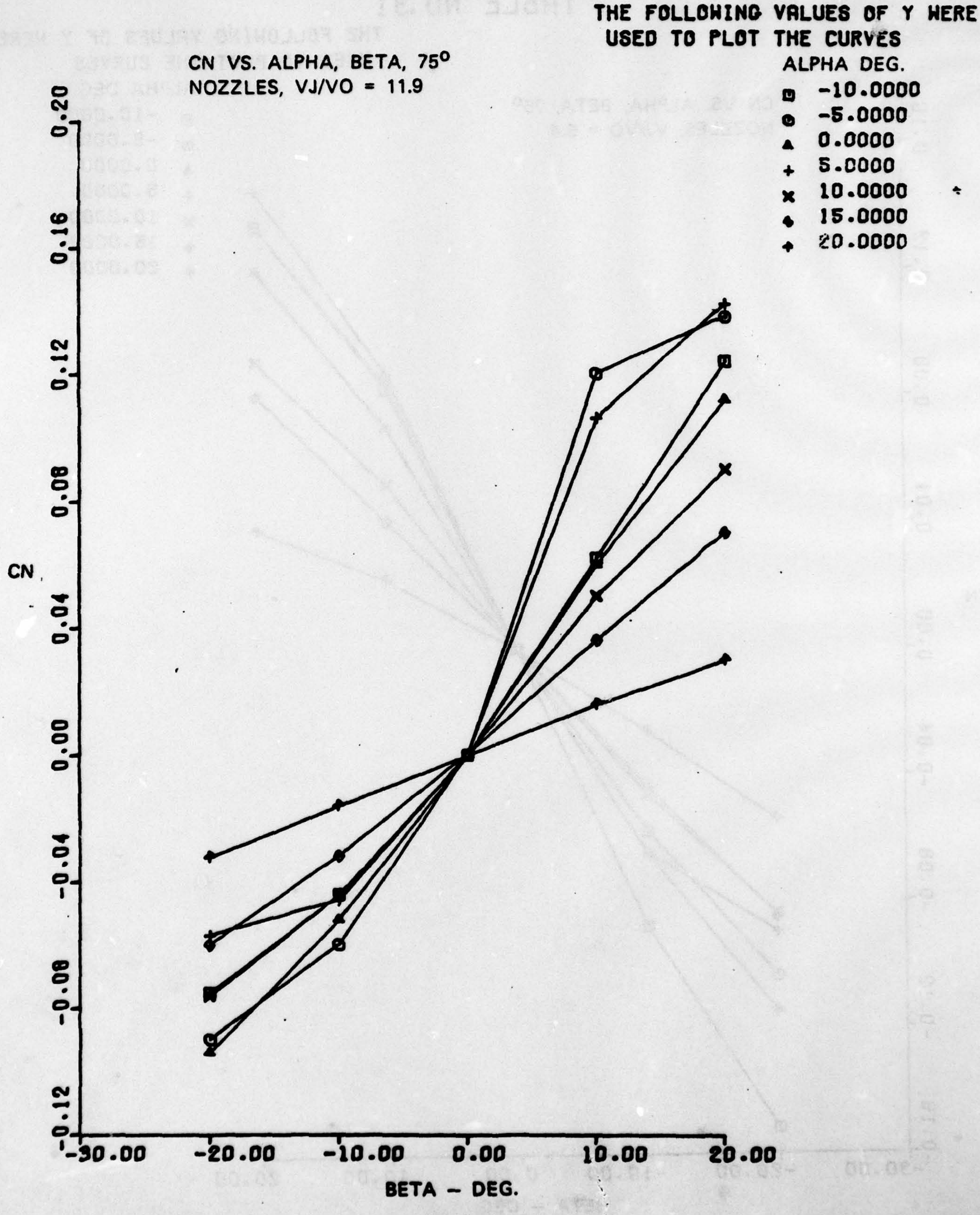

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
ALPHA DEG.
-10.0000
-5.0000
0.0000
5.0000
10.0000
15.0000
20.0000
CN VS. ALPHA, BETA, 75°
NOZZLES, VJ/VO = 11.9
CN
0.20
0.16
0.12
0.08
0.04
0.00
-0.04
-0.08
-0.12
-30.00
-20.00
-10.00
0.00
10.00
20.00
BETA - DEG.

TABLE NO.33

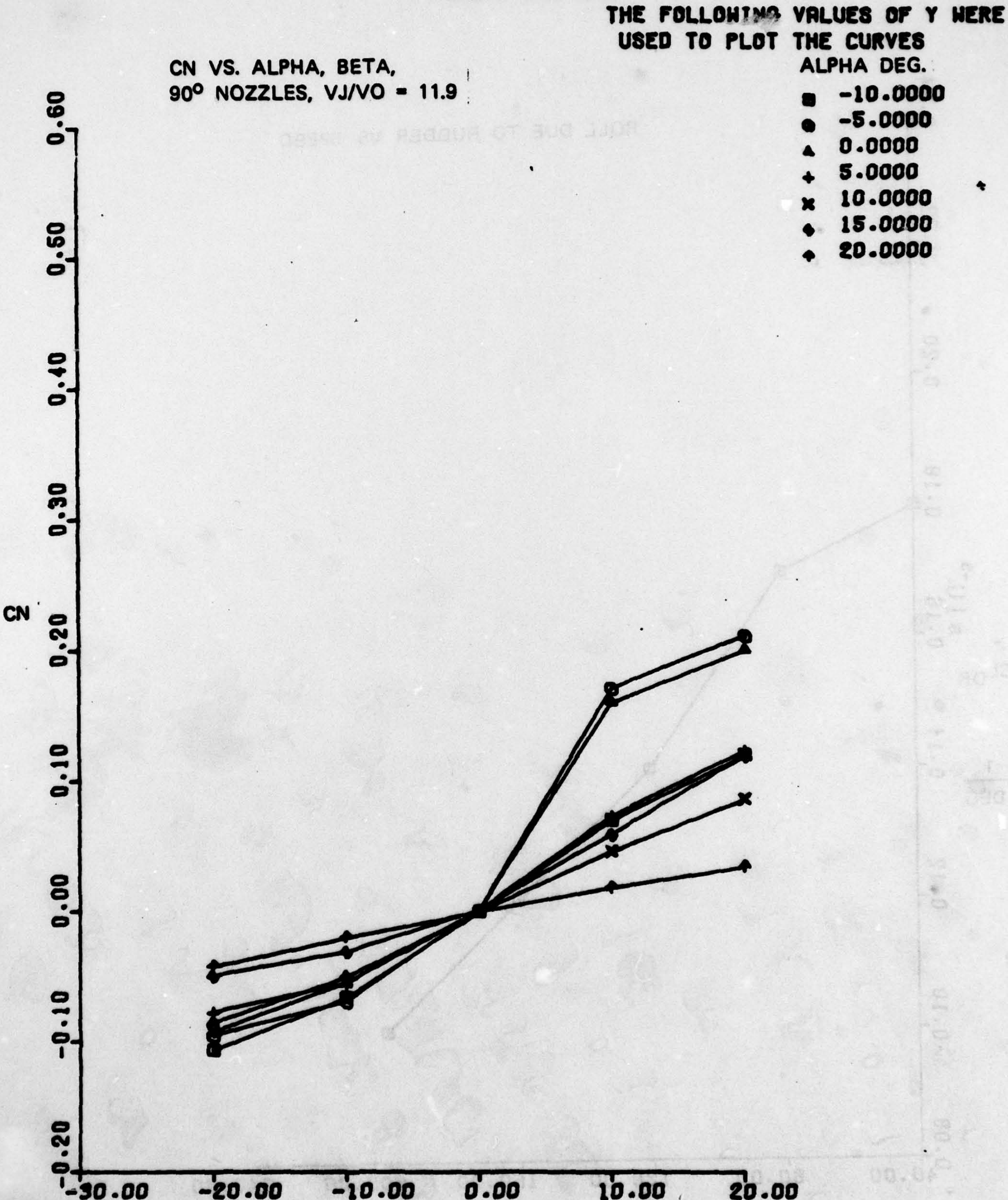

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
ALPHA DEG.
-10.0000
-5.0000
0.0000
5.0000
10.0000
15.0000
20.0000
CN VS. ALPHA, BETA,
90° NOZZLES, VJ/VO = 11.9
CN
0.60
0.50
0.40
0.30
0.20
0.10
0.00
-0.10
-0.20
-30.00
-20.00
-10.00
0.00
10.00
20.00
BETA

TABLE NO.34

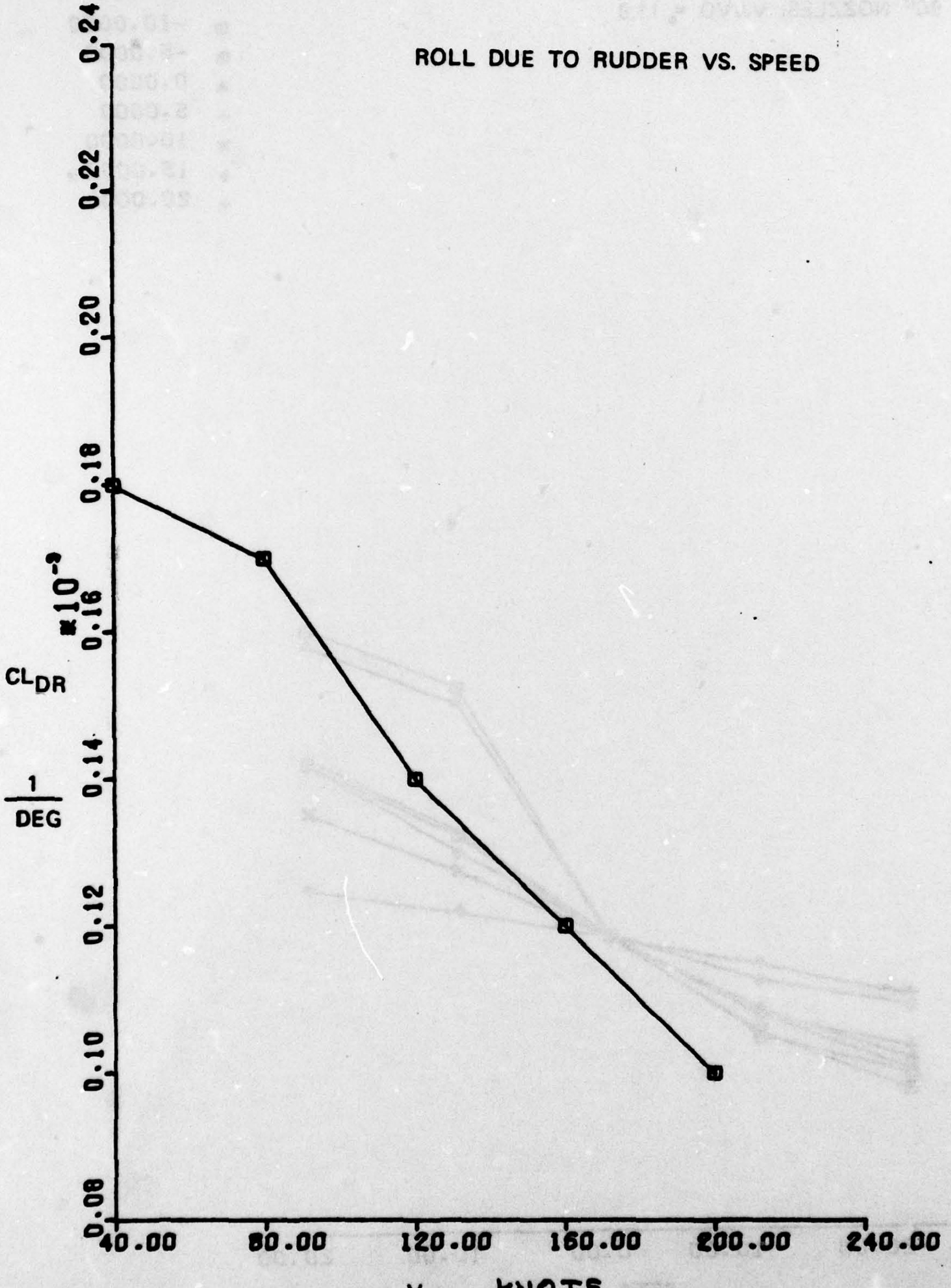

ROLL DUE TO RUDDER VS. SPEED
0.24
0.22
0.20
0.18
0.16
0.14
0.12
0.10
0.08
CL_{DR}
$\frac{1}{DEG}$
$\times 10^{-3}$
40.00
80.00
120.00
160.00
200.00
240.00
V - KNOTS

TABLE NO.35

YAWING COEFFICIENT PER UNIT
AILERON VS. ALPHA

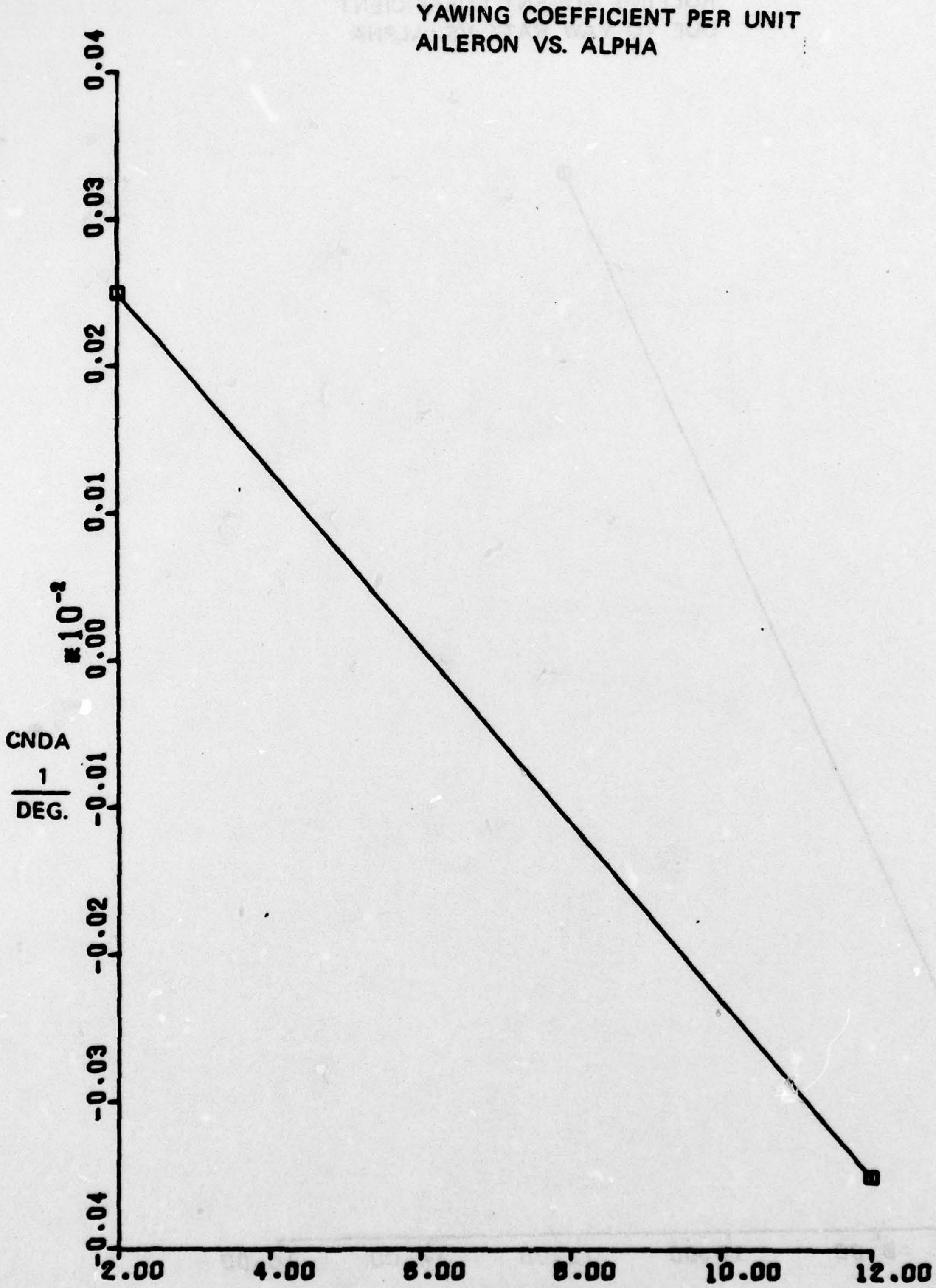


TABLE NO.36

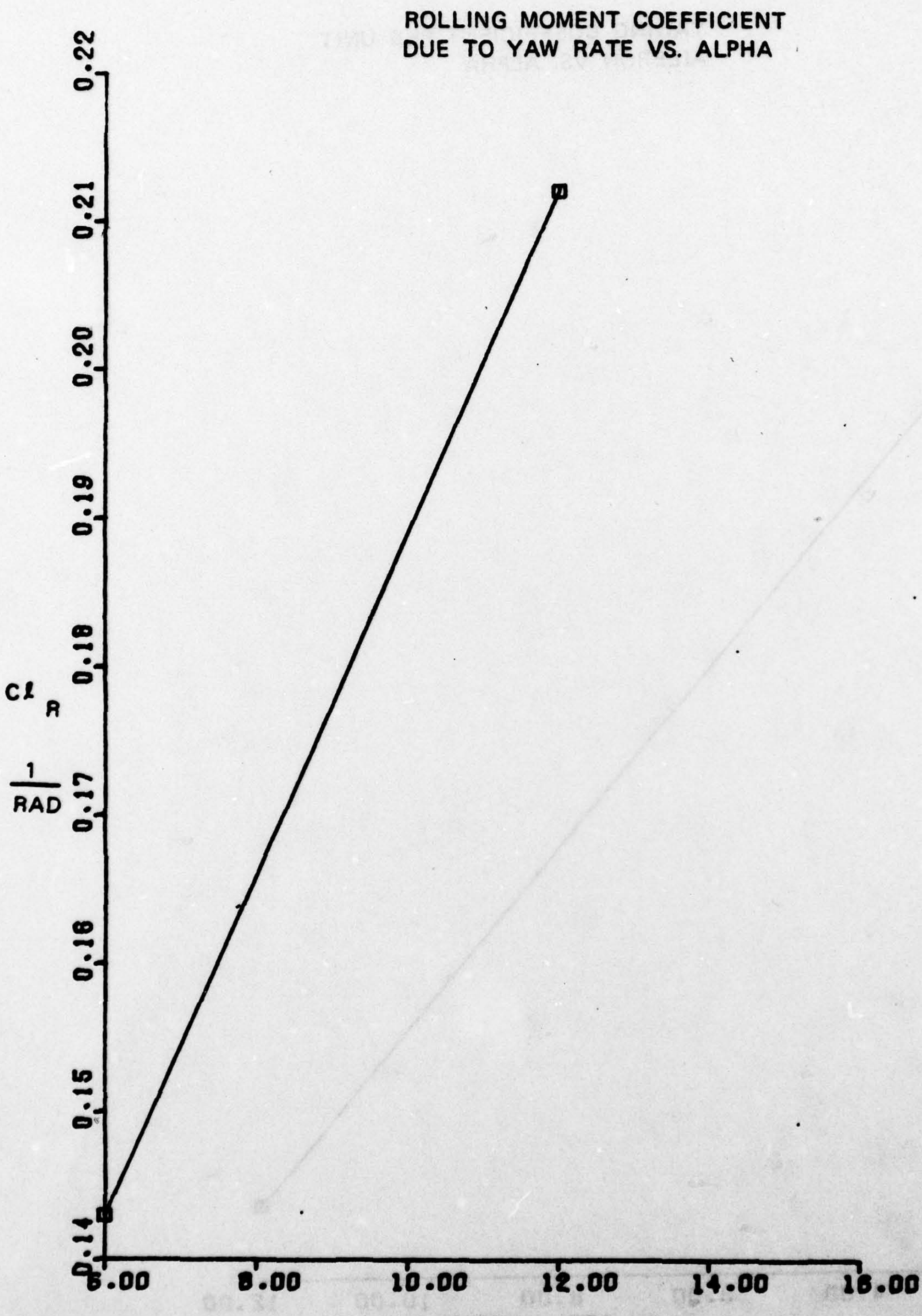

ROLLING MOMENT COEFFICIENT
DUE TO YAW RATE VS. ALPHA
0.22
0.21
0.20
0.19
0.18
0.17
0.16
0.15
0.14
$C\ell_R$
$\frac{1}{RAD}$
6.00
8.00
10.00
12.00
14.00
16.00
ALPHA - DEG.

TABLE NO.37

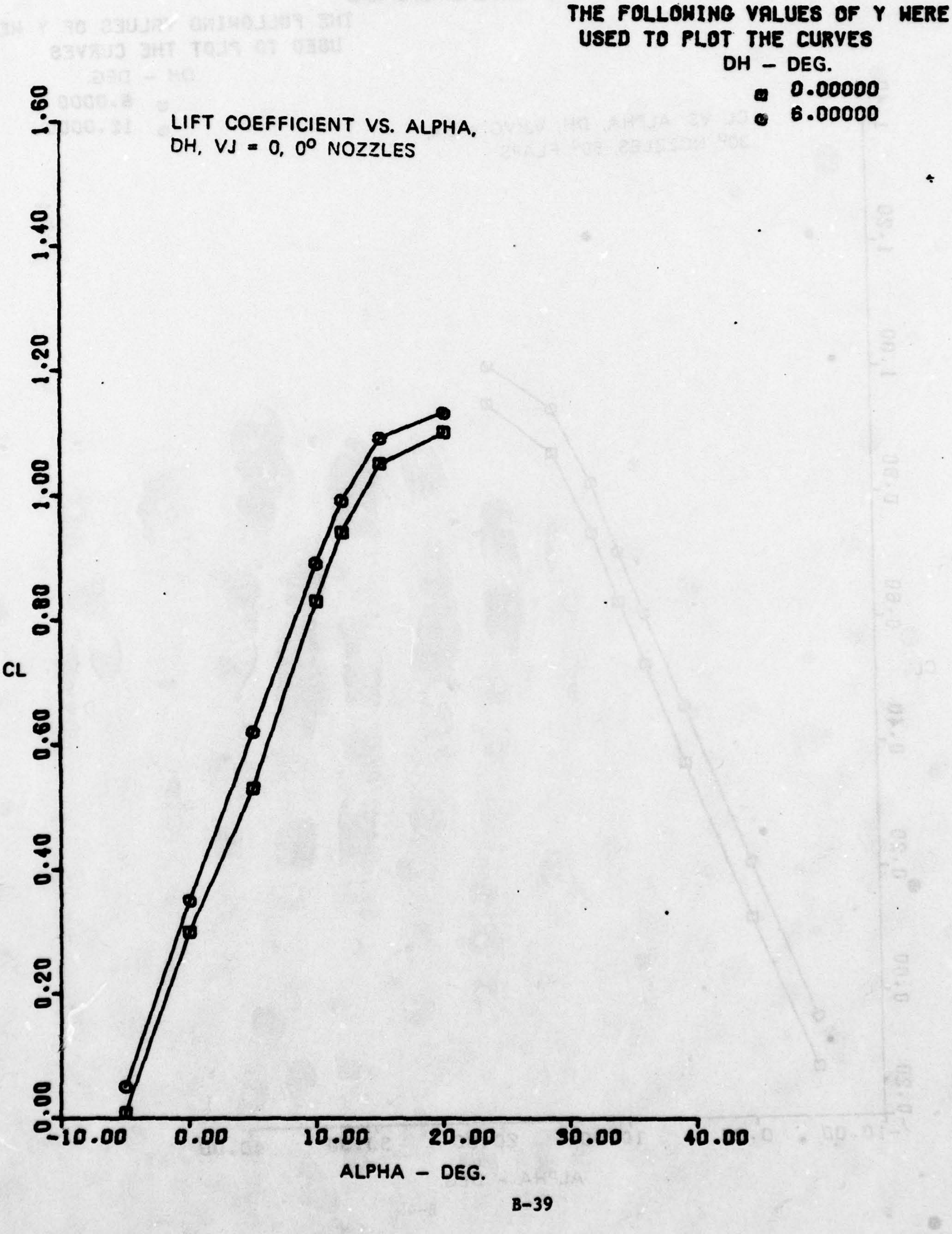

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
DH – DEG.
0.00000
6.00000
LIFT COEFFICIENT VS. ALPHA, DH, VJ = 0, 0° NOZZLES
CL
1.60
1.40
1.20
1.00
0.80
0.60
0.40
0.20
0.00
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA – DEG.

TABLE NO 38

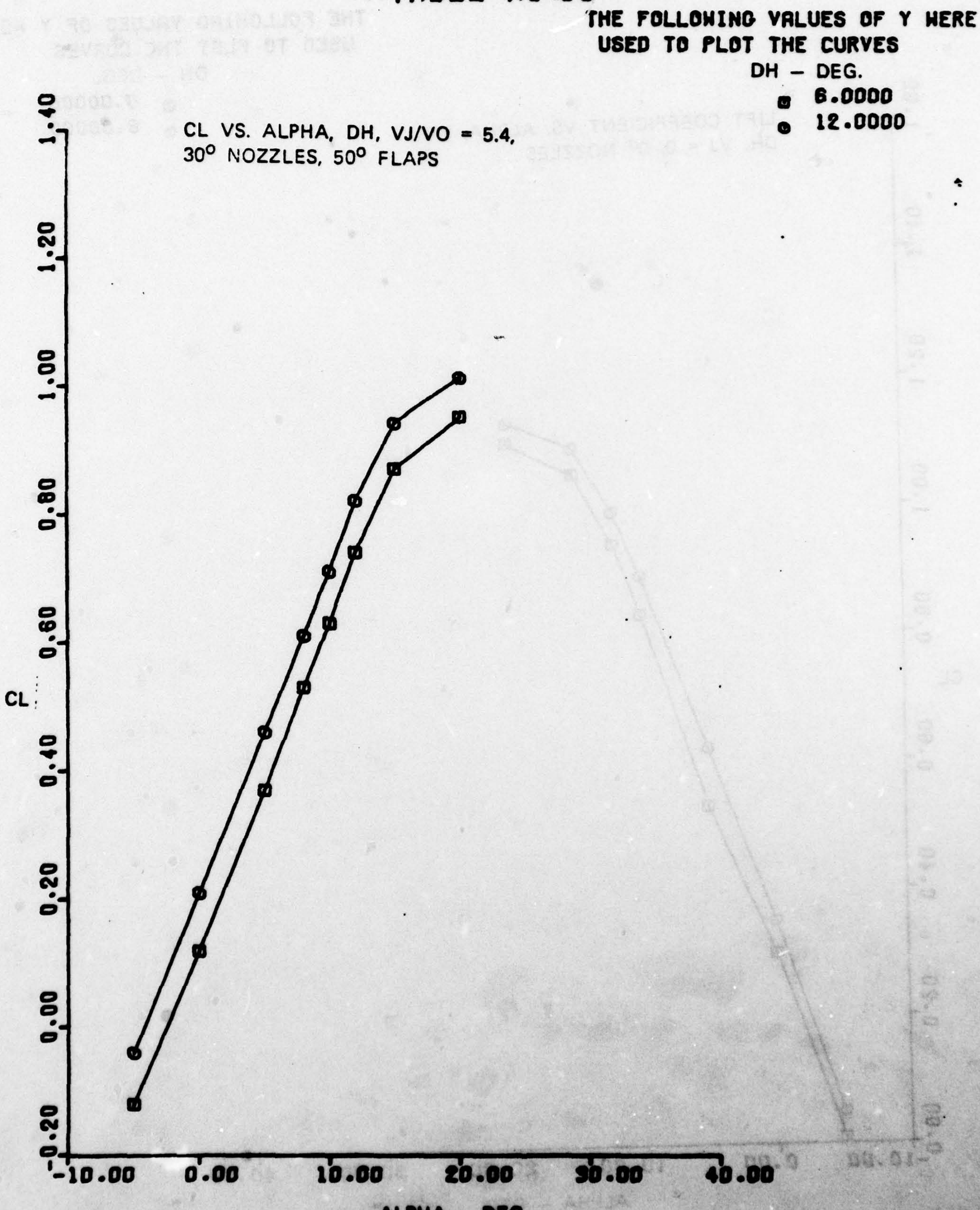

THE FOLLOWING VALUES OF Y WERE
USED TO PLOT THE CURVES
DH – DEG.
6.0000
12.0000
CL VS. ALPHA, DH, VJ/VO = 5.4,
30° NOZZLES, 50° FLAPS
CL
ALPHA – DEG.

TABLE NO 39

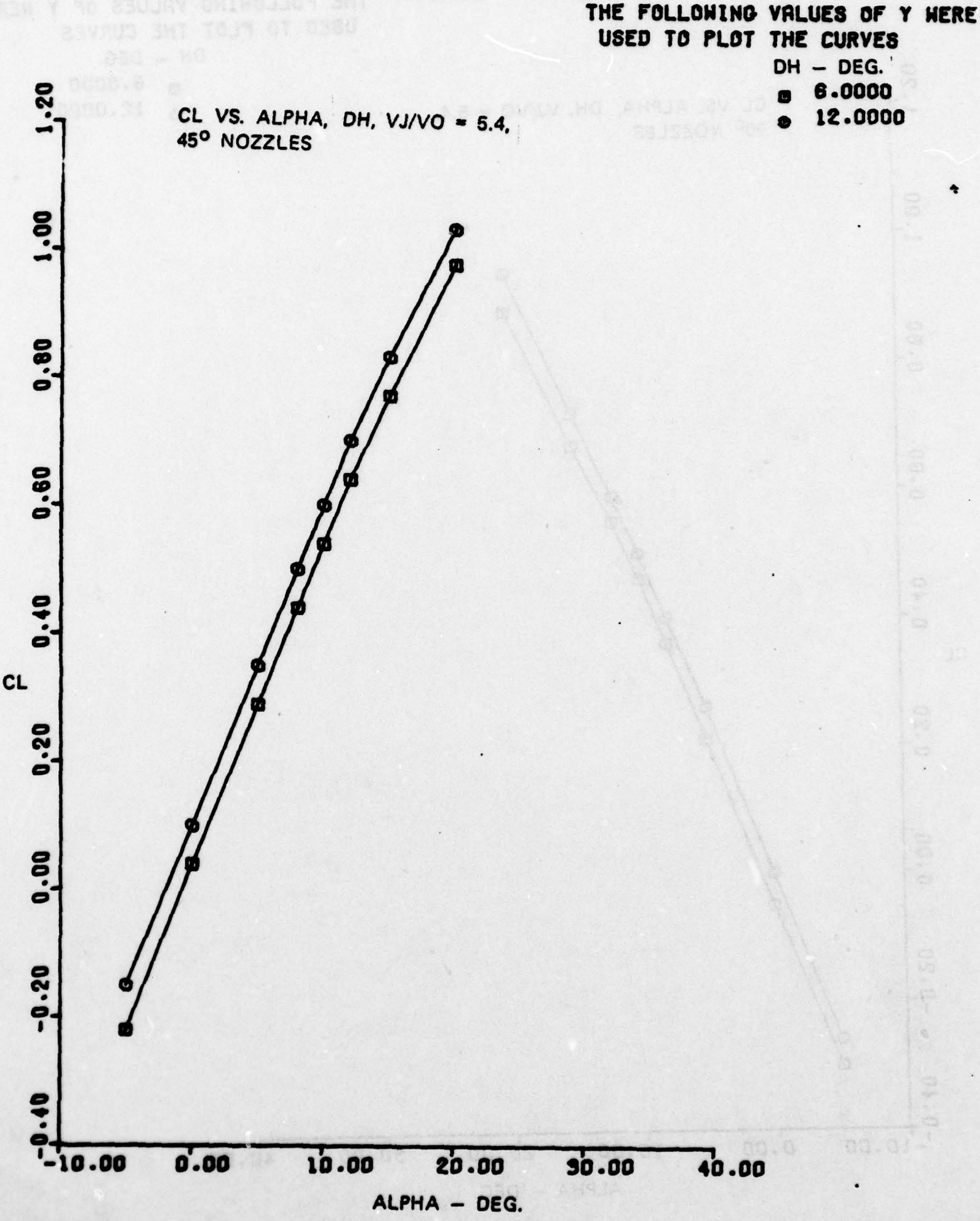

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
DH – DEG.
6.0000
12.0000
CL VS. ALPHA, DH, VJ/VO = 5.4, 45° NOZZLES
CL
1.20
1.00
0.80
0.60
0.40
0.20
0.00
-0.20
-0.40
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA – DEG.

TABLE NO.40

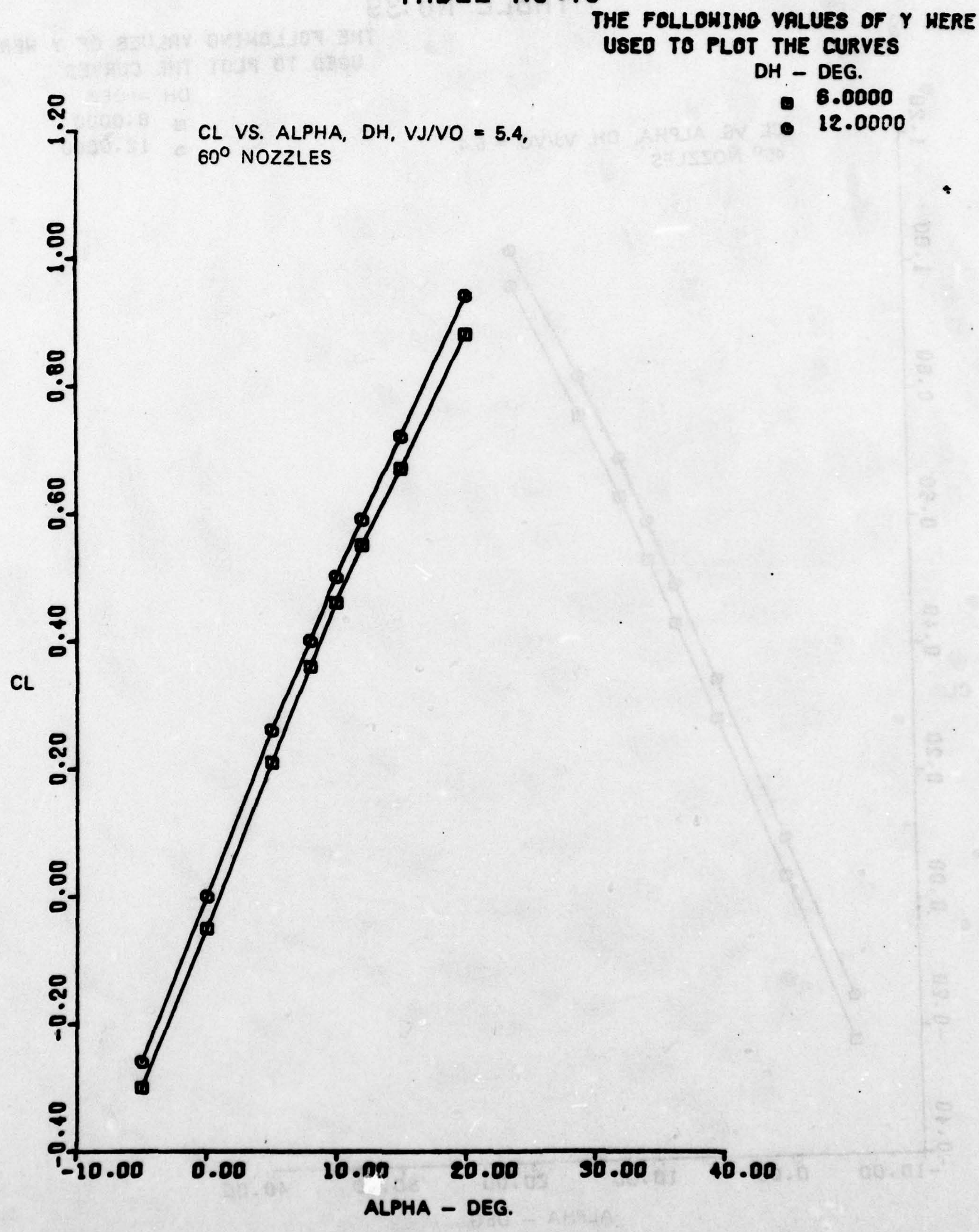

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
DH – DEG.
6.0000
12.0000
CL VS. ALPHA, DH, VJ/VO = 5.4,
60° NOZZLES
CL
1.20
1.00
0.80
0.60
0.40
0.20
0.00
-0.20
-0.40
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA – DEG.

TABLE NO.41

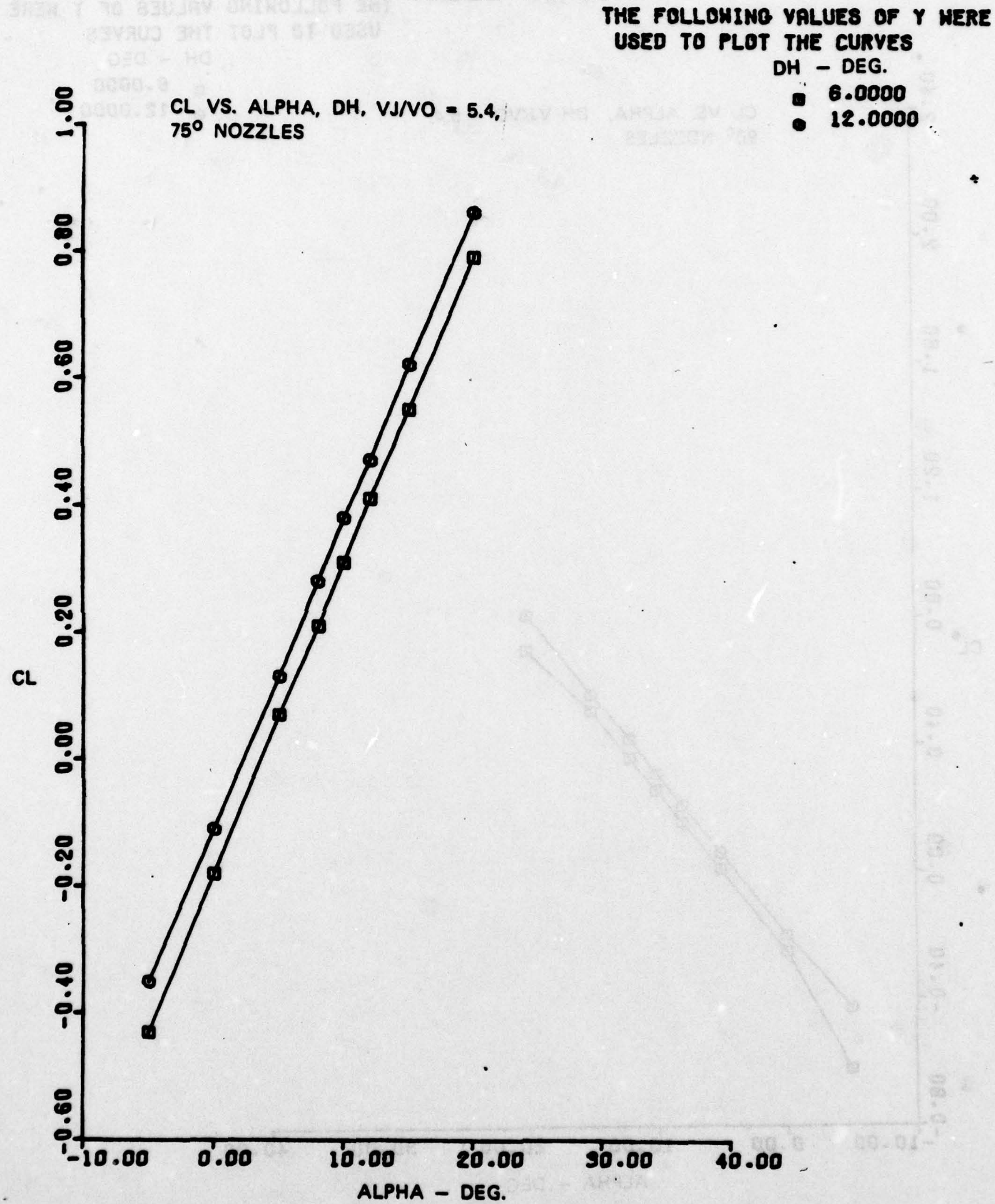

THE FOLLOWING VALUES OF Y WERE
USED TO PLOT THE CURVES
DH – DEG.
6.0000
12.0000
CL VS. ALPHA, DH, VJ/VO = 5.4,
75° NOZZLES
CL
1.00
0.80
0.60
0.40
0.20
0.00
-0.20
-0.40
-0.60
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA – DEG.

TABLE NO.42

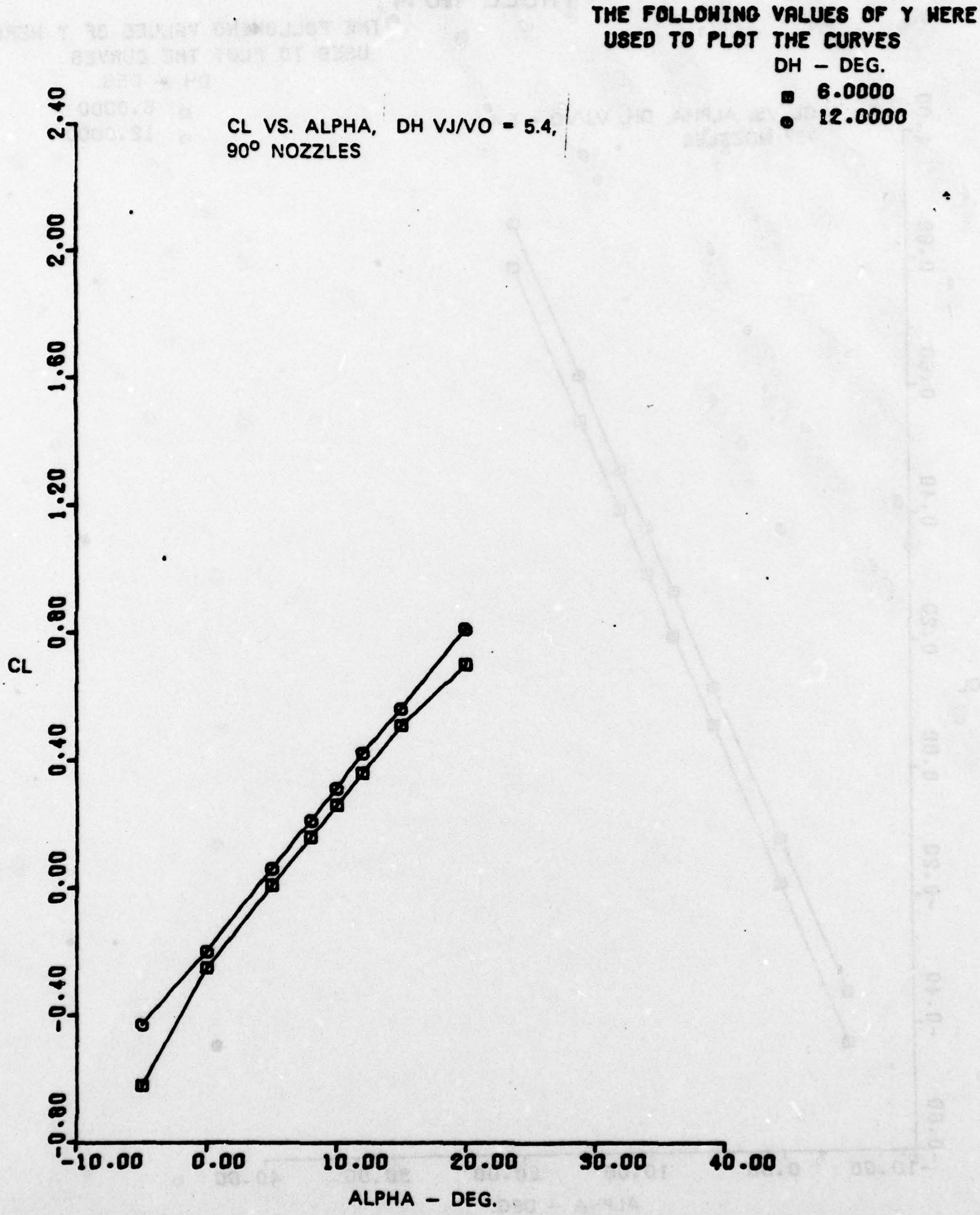

THE FOLLOWING VALUES OF Y WERE
USED TO PLOT THE CURVES
DH – DEG.
6.0000
12.0000
CL VS. ALPHA, DH VJ/VO = 5.4,
90° NOZZLES
CL
2.40
2.00
1.60
1.20
0.80
0.40
0.00
-0.40
-0.80
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA – DEG.

TABLE NO.43

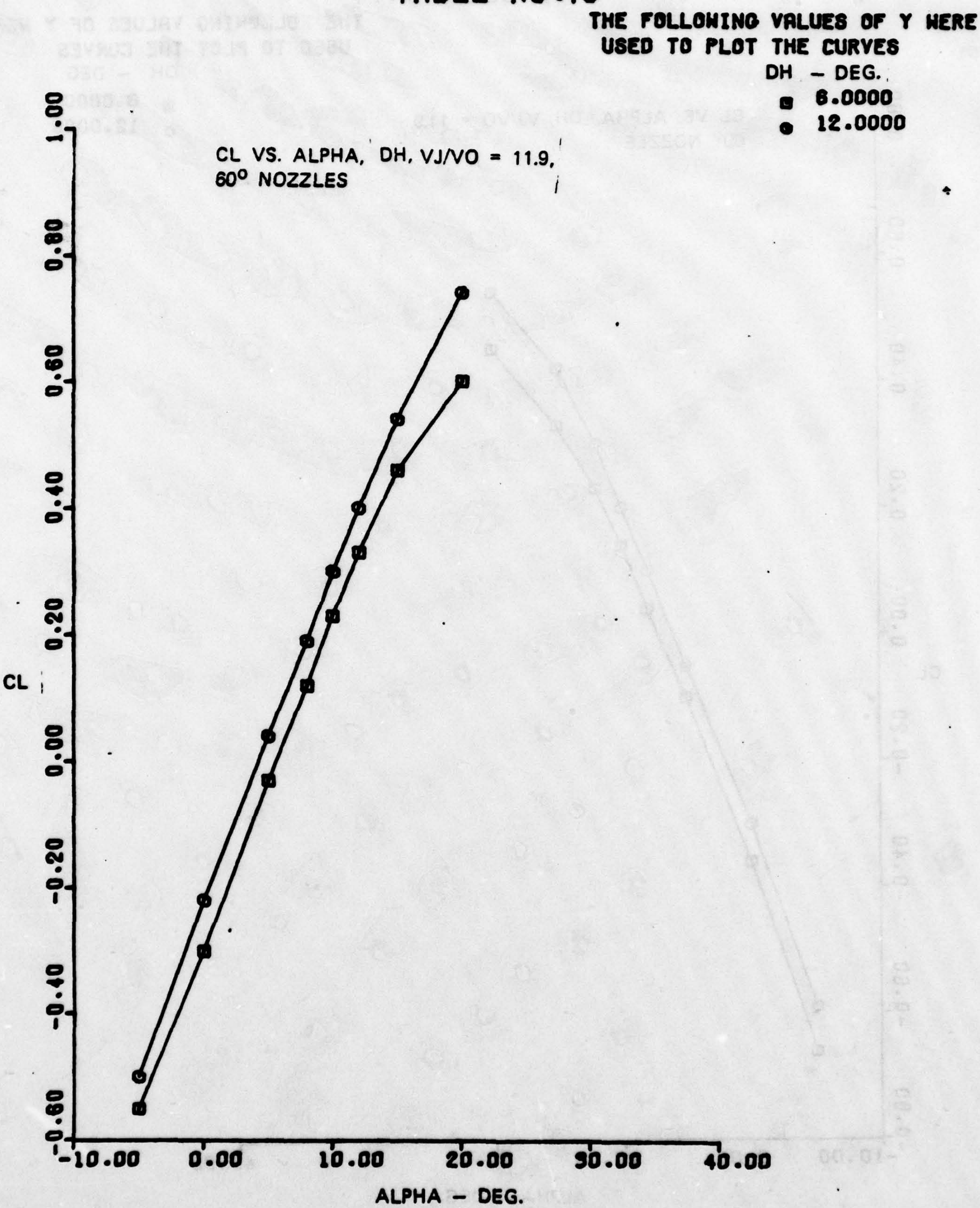

THE FOLLOWING VALUES OF Y WERE
USED TO PLOT THE CURVES
DH - DEG.
6.0000
12.0000
CL VS. ALPHA, DH, VJ/VO = 11.9,
60° NOZZLES
1.00
0.80
0.60
0.40
0.20
0.00
-0.20
-0.40
-0.60
CL
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA - DEG.

TABLE NO.44

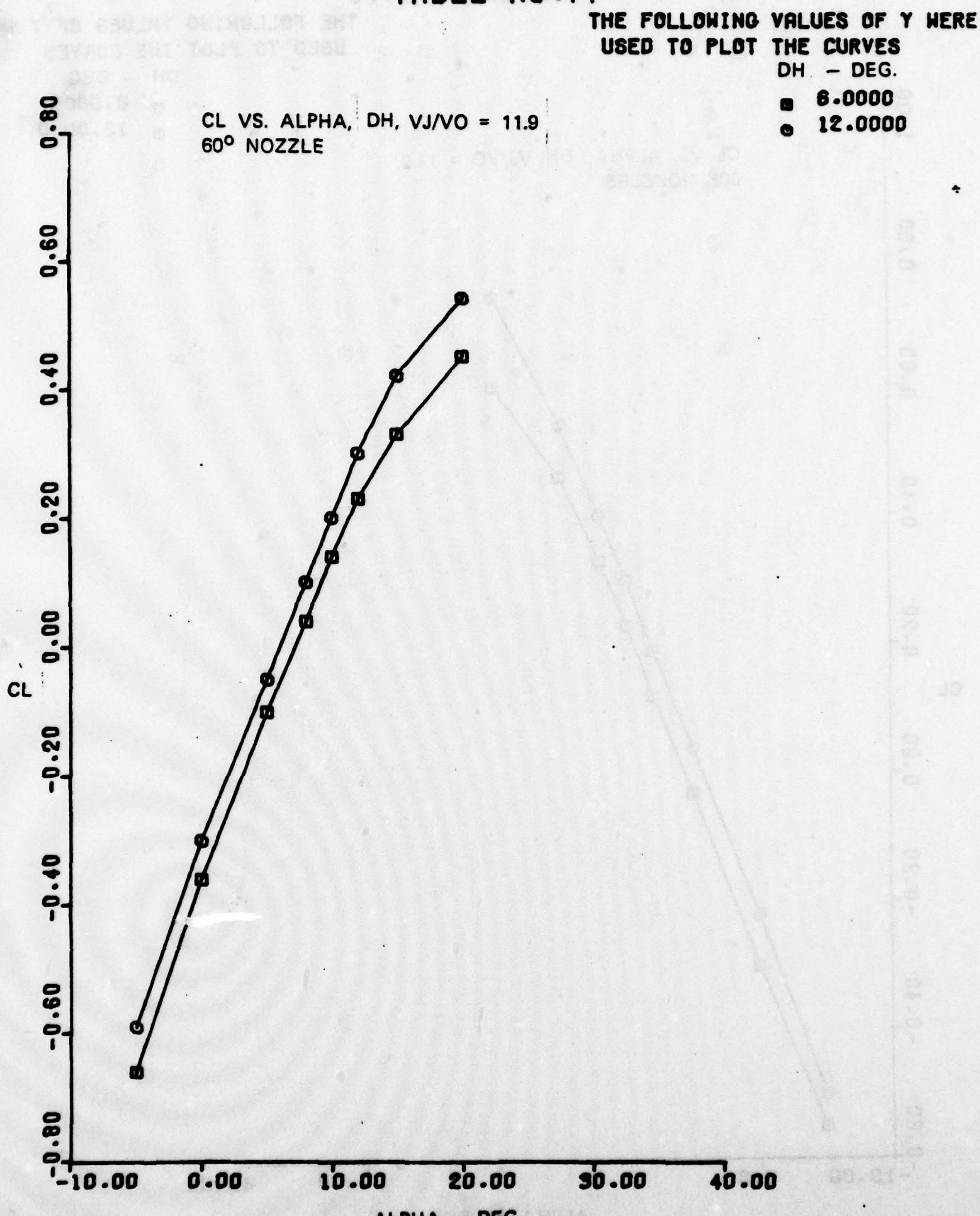

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
DH - DEG.
6.0000
12.0000
CL VS. ALPHA, DH, VJ/VO = 11.9
60° NOZZLE
CL
0.80
0.60
0.40
0.20
0.00
-0.20
-0.40
-0.60
-0.80
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA - DEG.

TABLE NO.45

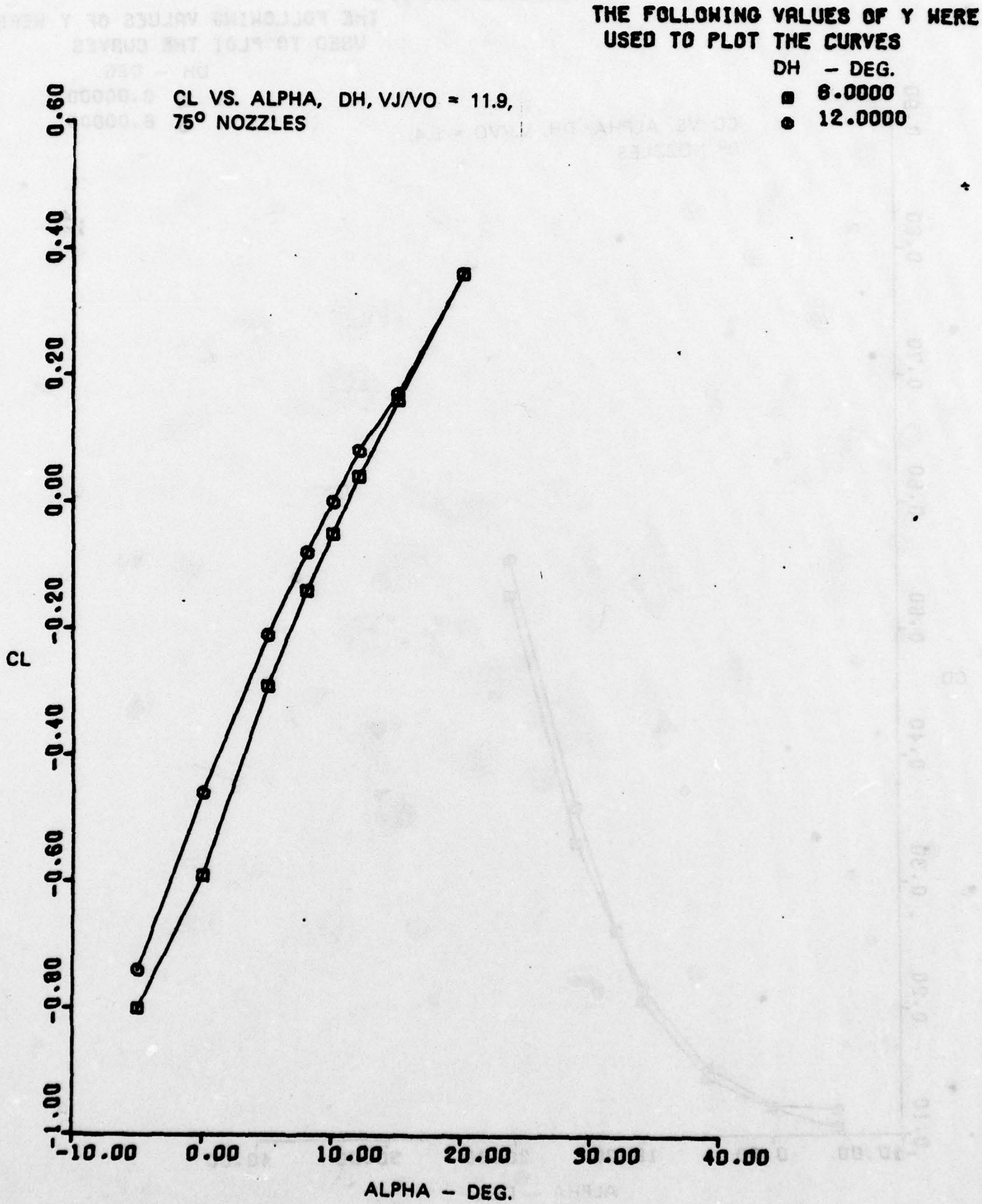

THE FOLLOWING VALUES OF Y WERE
USED TO PLOT THE CURVES
DH - DEG.
6.0000
12.0000
CL VS. ALPHA, DH, VJ/VO = 11.9,
75° NOZZLES
0.60
0.40
0.20
0.00
-0.20
-0.40
-0.60
-0.80
-1.00
CL
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA - DEG.

TABLE NO.46

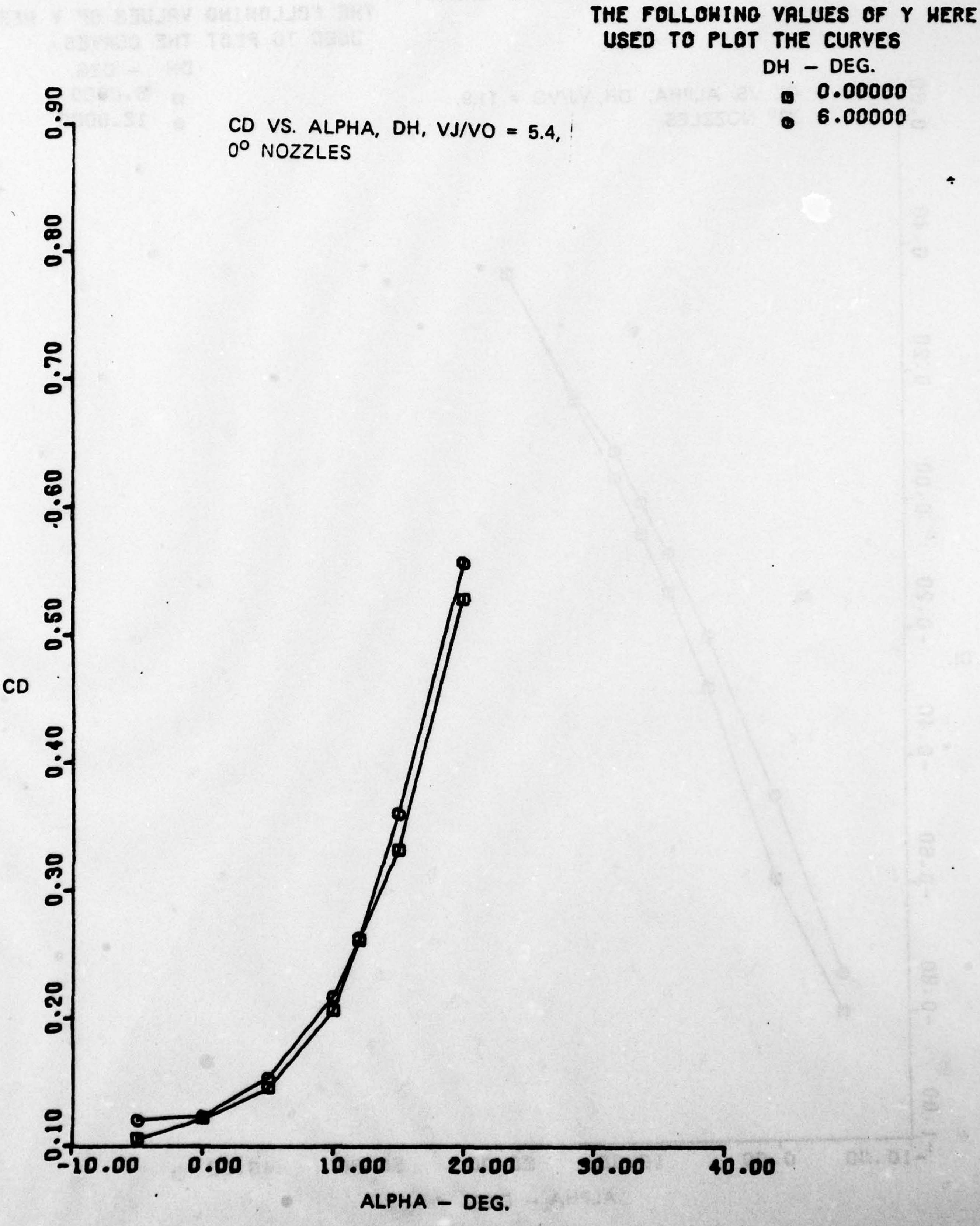

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
DH – DEG.
0.00000
6.00000
CD VS. ALPHA, DH, VJ/VO = 5.4,
0° NOZZLES
CD
0.10
0.20
0.30
0.40
0.50
0.60
0.70
0.80
0.90
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA – DEG.

TABLE NO.47

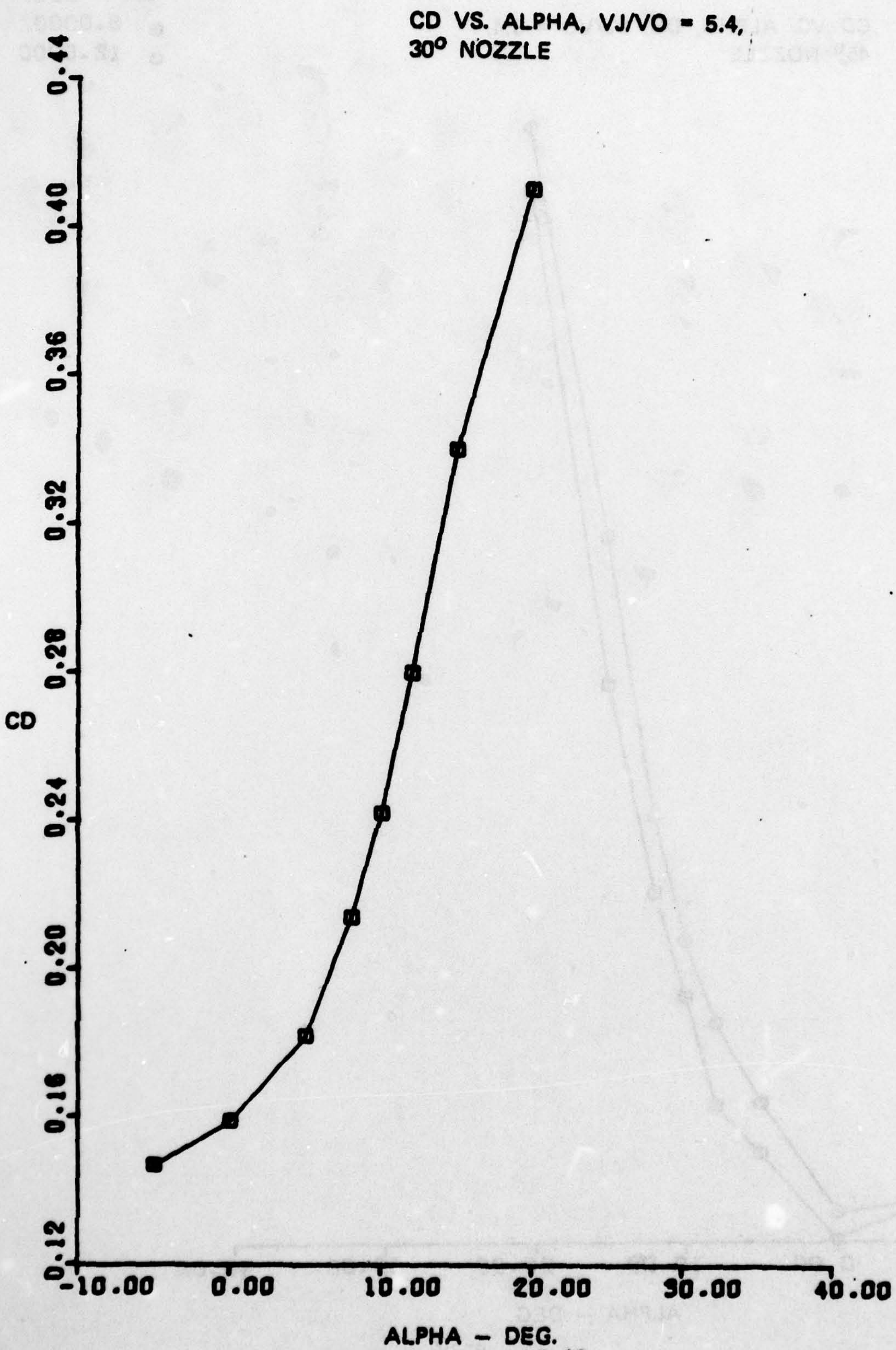

CD VS. ALPHA, VJ/VO = 5.4,
30° NOZZLE
CD
0.12
0.16
0.20
0.24
0.28
0.32
0.36
0.40
0.44
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA - DEG.

TABLE NO.48

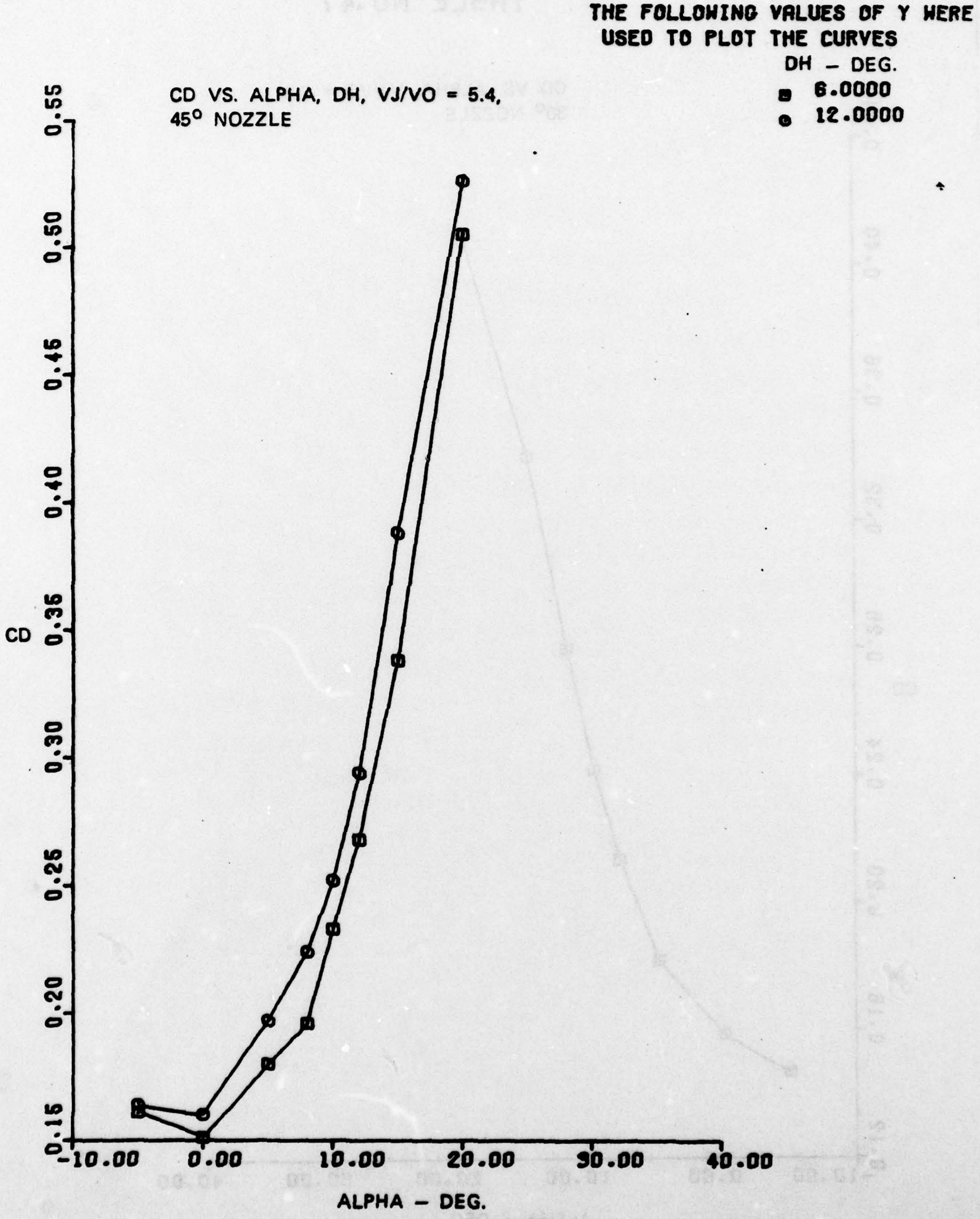

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
DH – DEG.
6.0000
12.0000
CD VS. ALPHA, DH, VJ/VO = 5.4,
45° NOZZLE
CD
0.15
0.20
0.25
0.30
0.35
0.40
0.45
0.50
0.55
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA – DEG.

TABLE NO.49

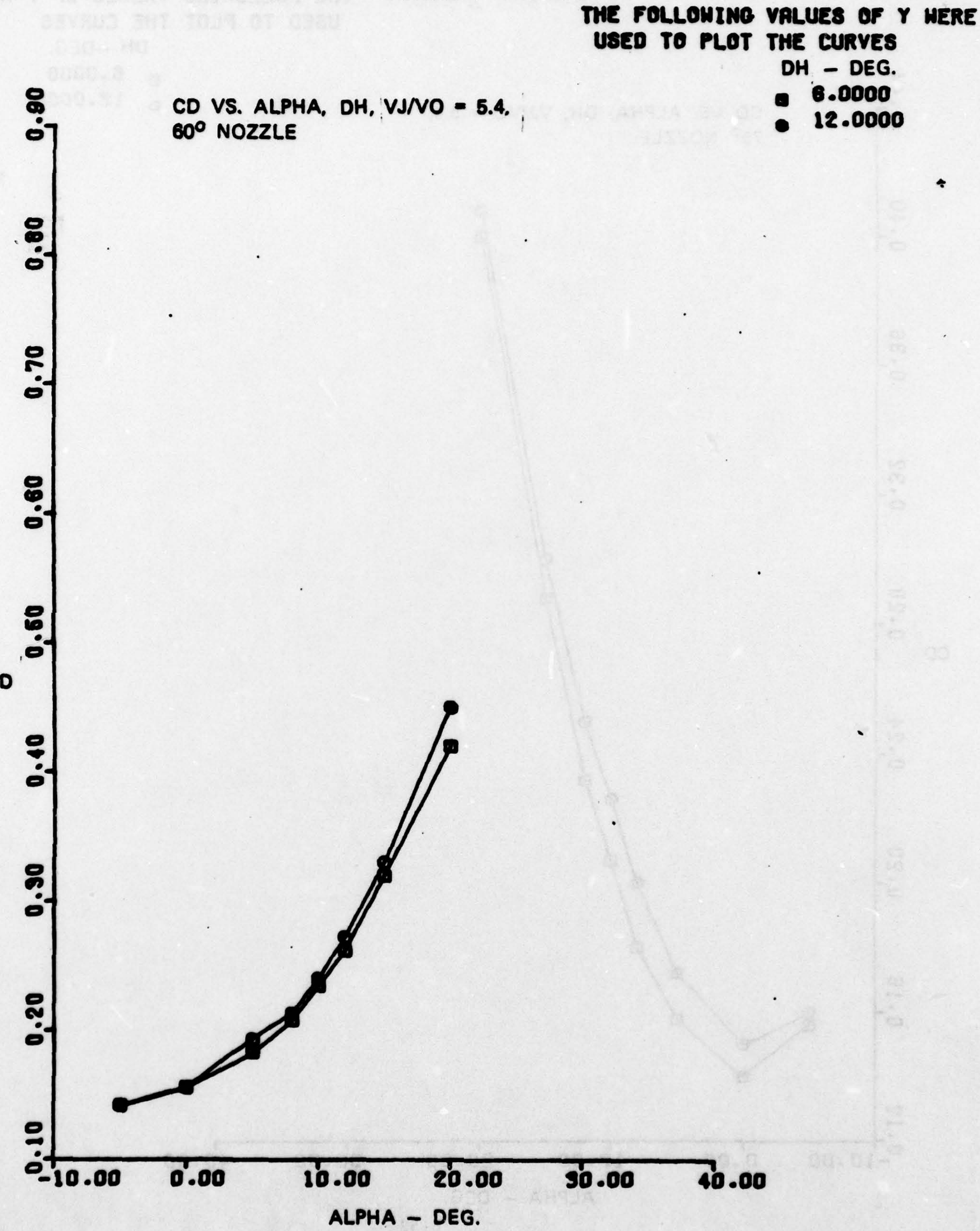

THE FOLLOWING VALUES OF Y WERE
USED TO PLOT THE CURVES
DH - DEG.
6.0000
12.0000
CD VS. ALPHA, DH, VJ/VO = 5.4.
60° NOZZLE
CD
0.10
0.20
0.30
0.40
0.50
0.60
0.70
0.80
0.90
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA - DEG.

TABLE NO 50

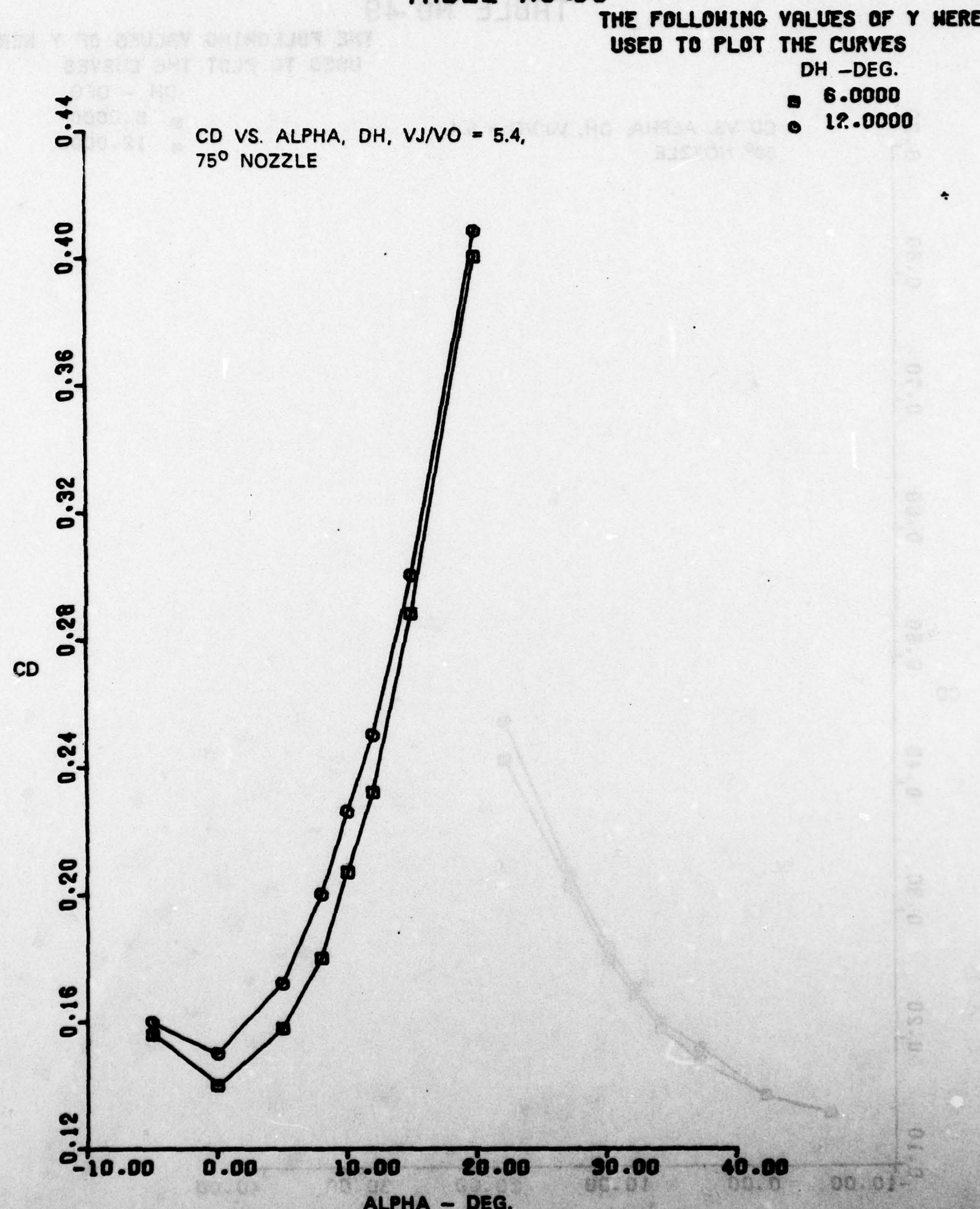

THE FOLLOWING VALUES OF Y WERE
USED TO PLOT THE CURVES
DH -DEG.
6.0000
12.0000
CD VS. ALPHA, DH, VJ/VO = 5.4,
75° NOZZLE
CD
0.12
0.16
0.20
0.24
0.28
0.32
0.36
0.40
0.44
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA - DEG.

TABLE NO.51

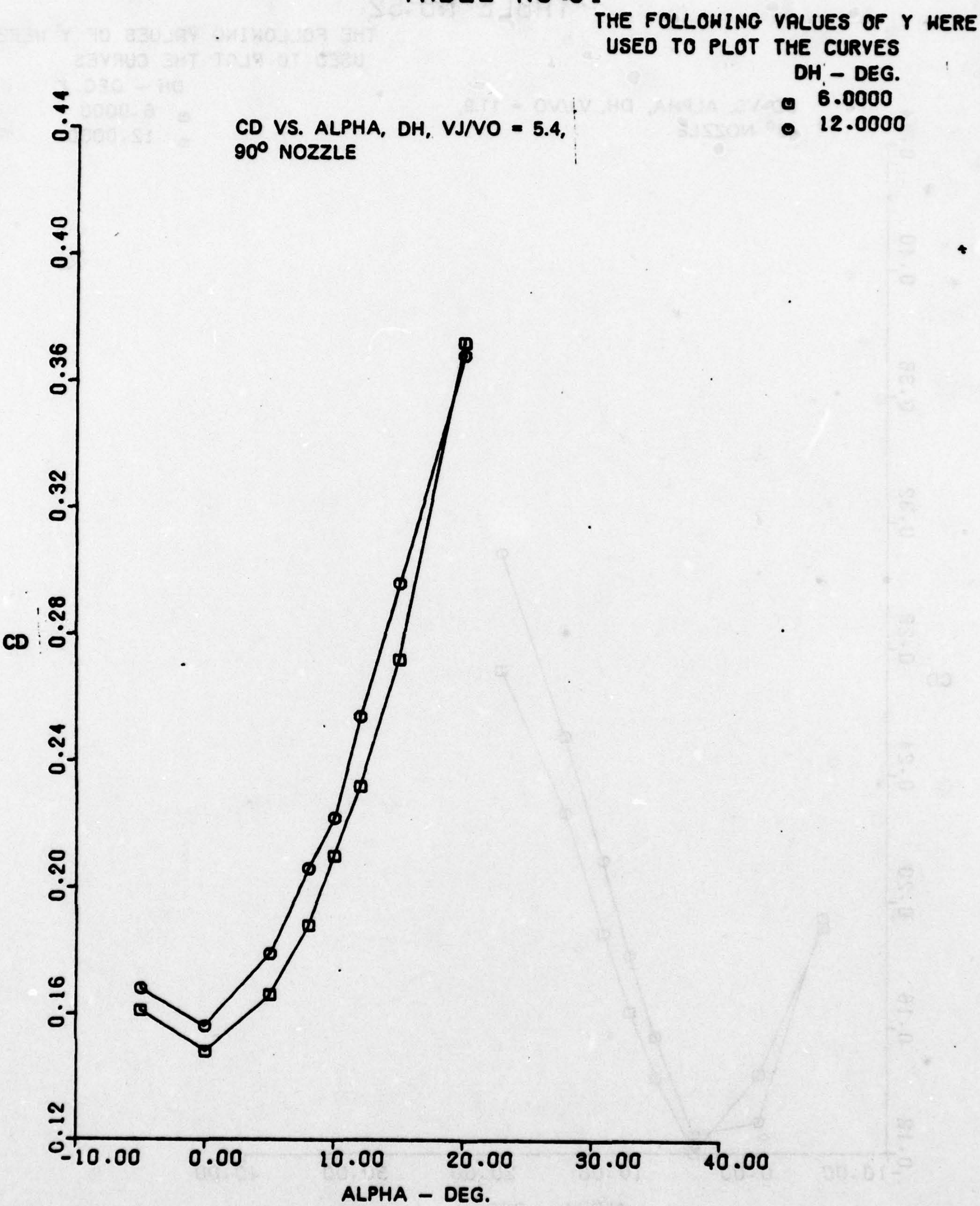

THE FOLLOWING VALUES OF Y WERE
USED TO PLOT THE CURVES
DH - DEG.
6.0000
12.0000
CD VS. ALPHA, DH, VJ/VO = 5.4,
90° NOZZLE
CD
0.12
0.16
0.20
0.24
0.28
0.32
0.36
0.40
0.44
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA - DEG.

TABLE NO 52

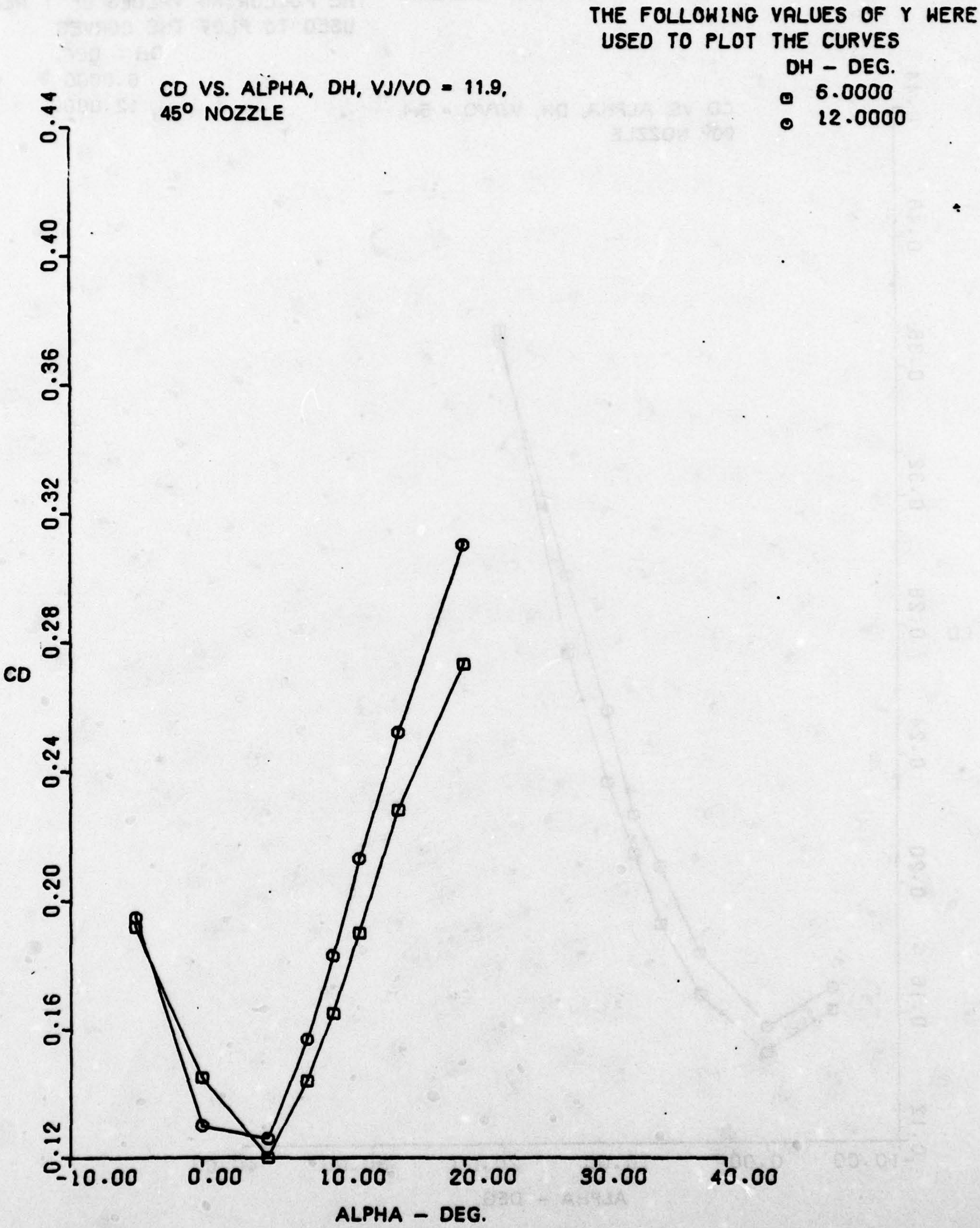

THE FOLLOWING VALUES OF Y WERE
USED TO PLOT THE CURVES
DH - DEG.
6.0000
12.0000
CD VS. ALPHA, DH, VJ/VO = 11.9,
45° NOZZLE
CD
0.12
0.16
0.20
0.24
0.28
0.32
0.36
0.40
0.44
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA - DEG.

TABLE NO.53

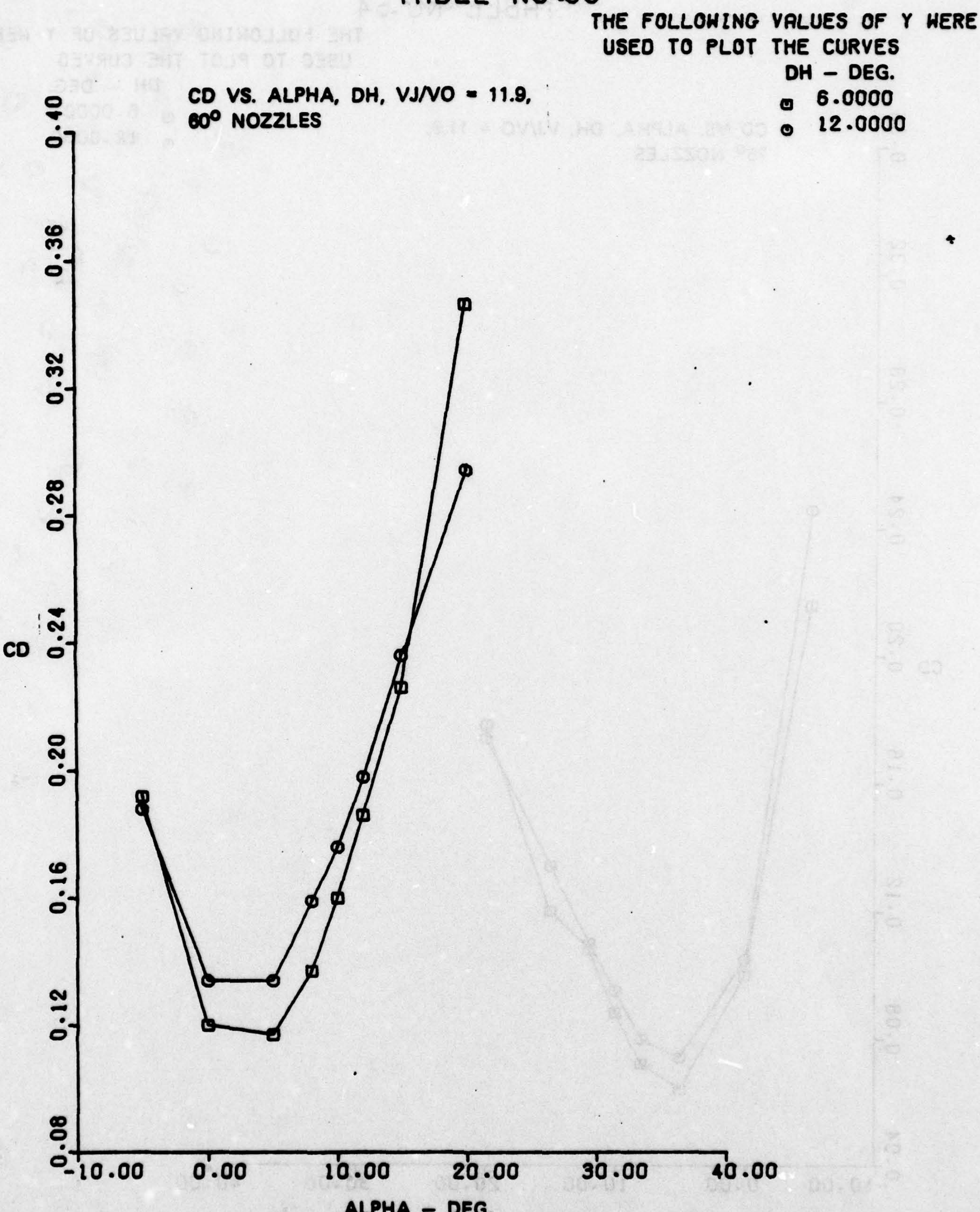

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
DH – DEG.
6.0000
12.0000
CD VS. ALPHA, DH, VJ/VO = 11.9, 60° NOZZLES
CD
0.08
0.12
0.16
0.20
0.24
0.28
0.32
0.36
0.40
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA – DEG.

TABLE NO.54

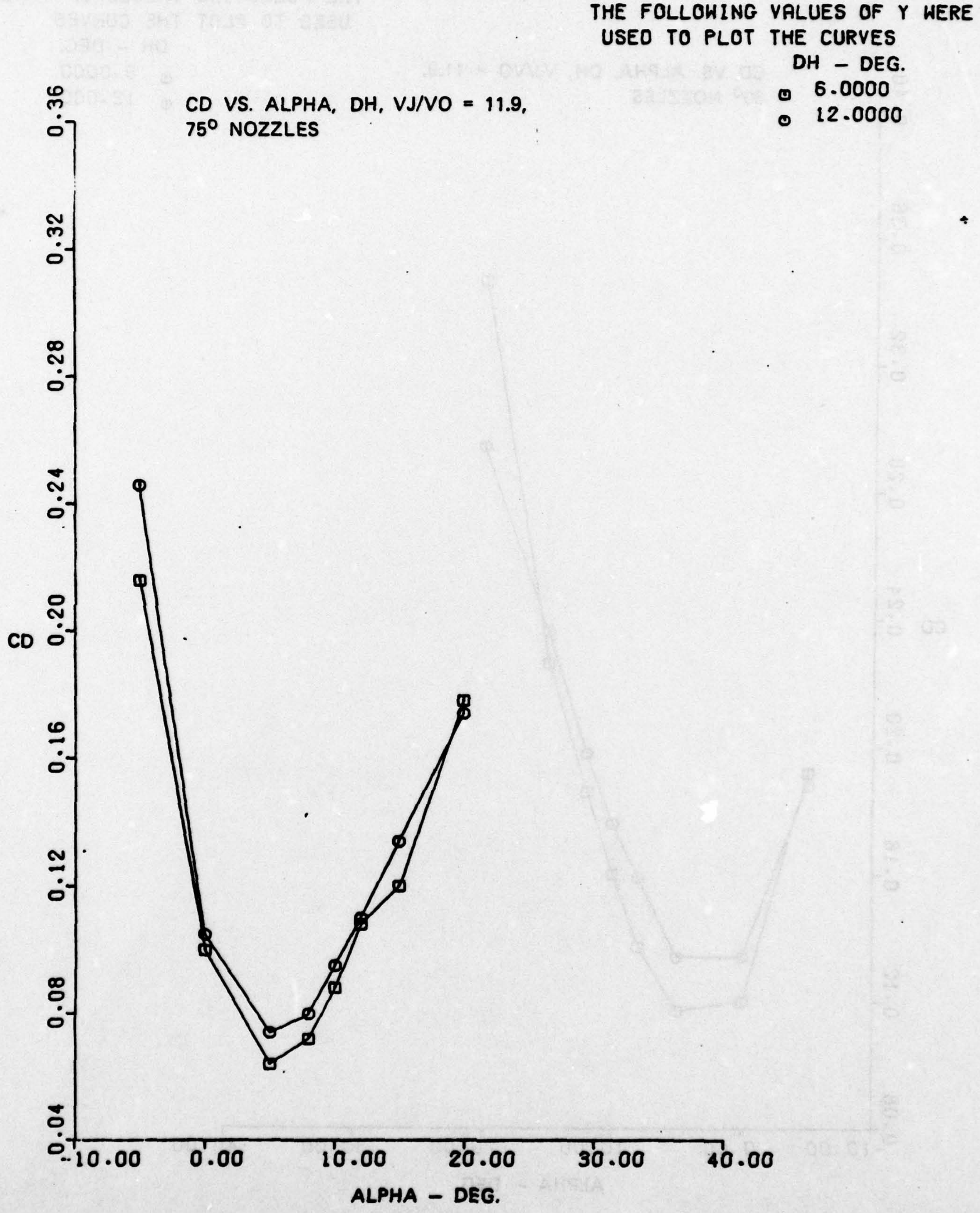

THE FOLLOWING VALUES OF Y WERE
USED TO PLOT THE CURVES
DH – DEG.
6.0000
12.0000
CD VS. ALPHA, DH, VJ/VO = 11.9,
75° NOZZLES
CD
0.04
0.08
0.12
0.16
0.20
0.24
0.28
0.32
0.36
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA – DEG.

TABLE NO.55

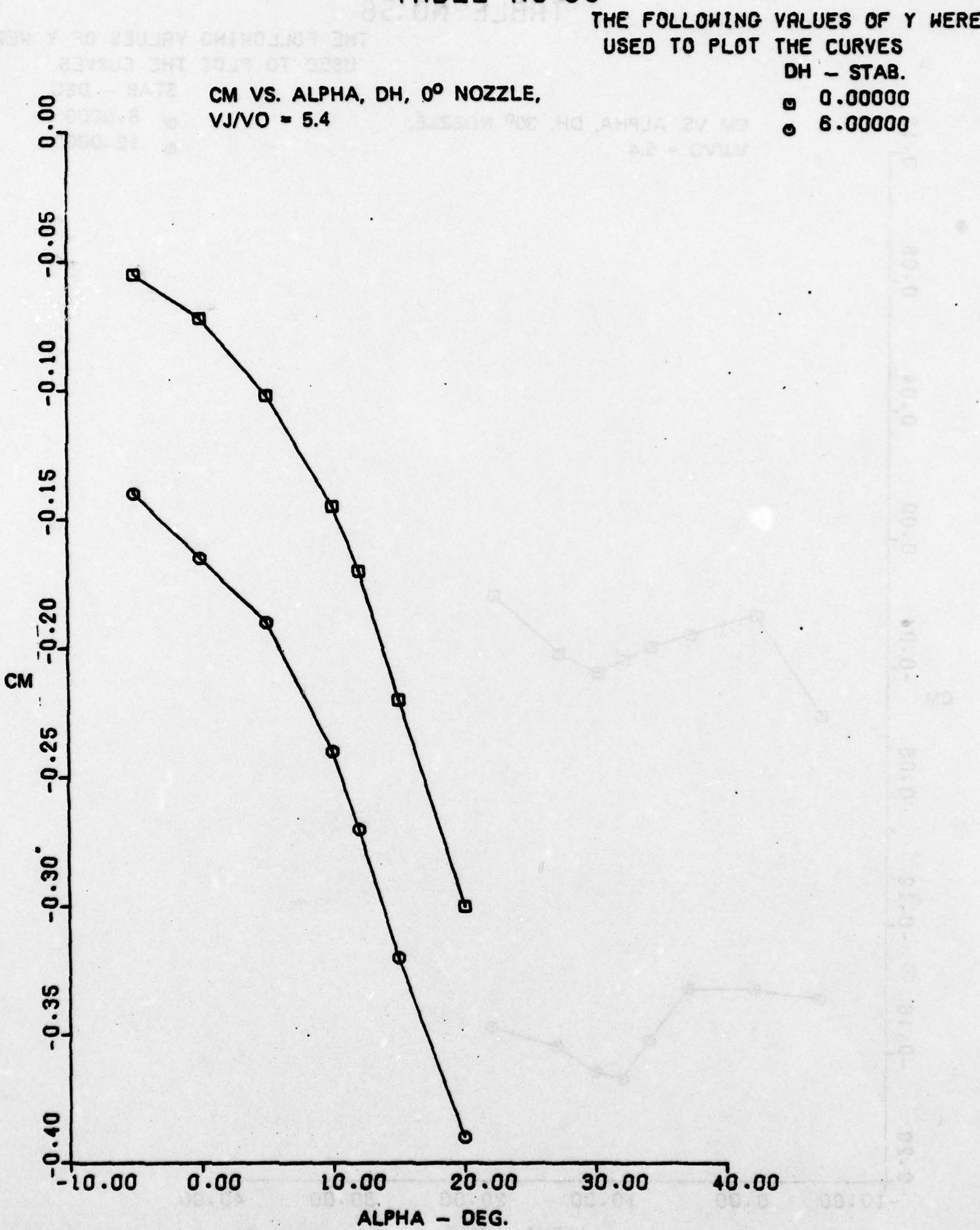

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
DH – STAB.
0.00000
6.00000
CM VS. ALPHA, DH, 0° NOZZLE,
VJ/VO = 5.4
CM
0.00
-0.05
-0.10
-0.15
-0.20
-0.25
-0.30
-0.35
-0.40
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA – DEG.

TABLE NO.56

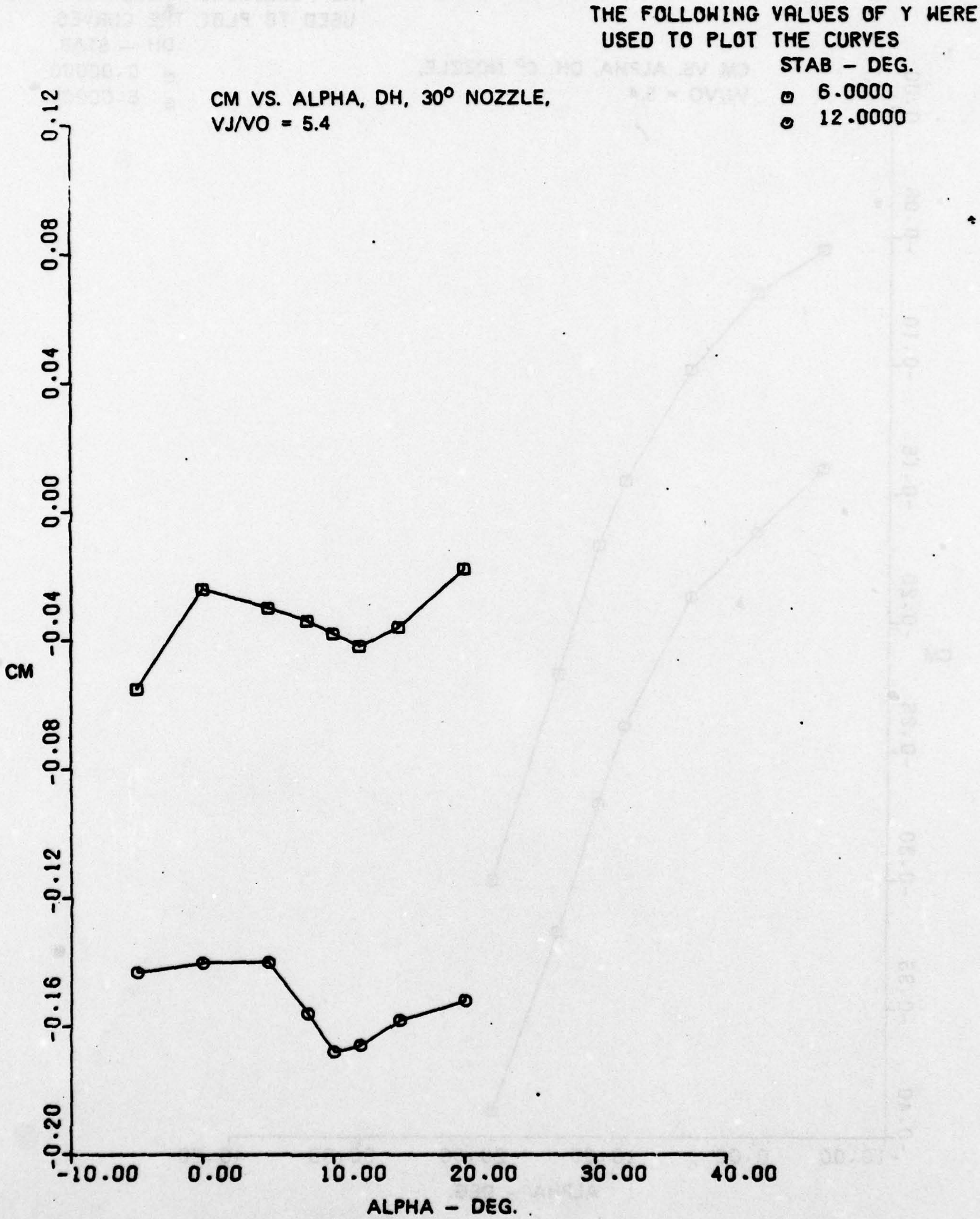

THE FOLLOWING VALUES OF Y WERE
USED TO PLOT THE CURVES
STAB - DEG.
6.0000
12.0000
CM VS. ALPHA, DH, 30° NOZZLE,
VJ/VO = 5.4
0.12
0.08
0.04
0.00
-0.04
-0.08
-0.12
-0.16
-0.20
CM
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA - DEG.

TABLE NO.57

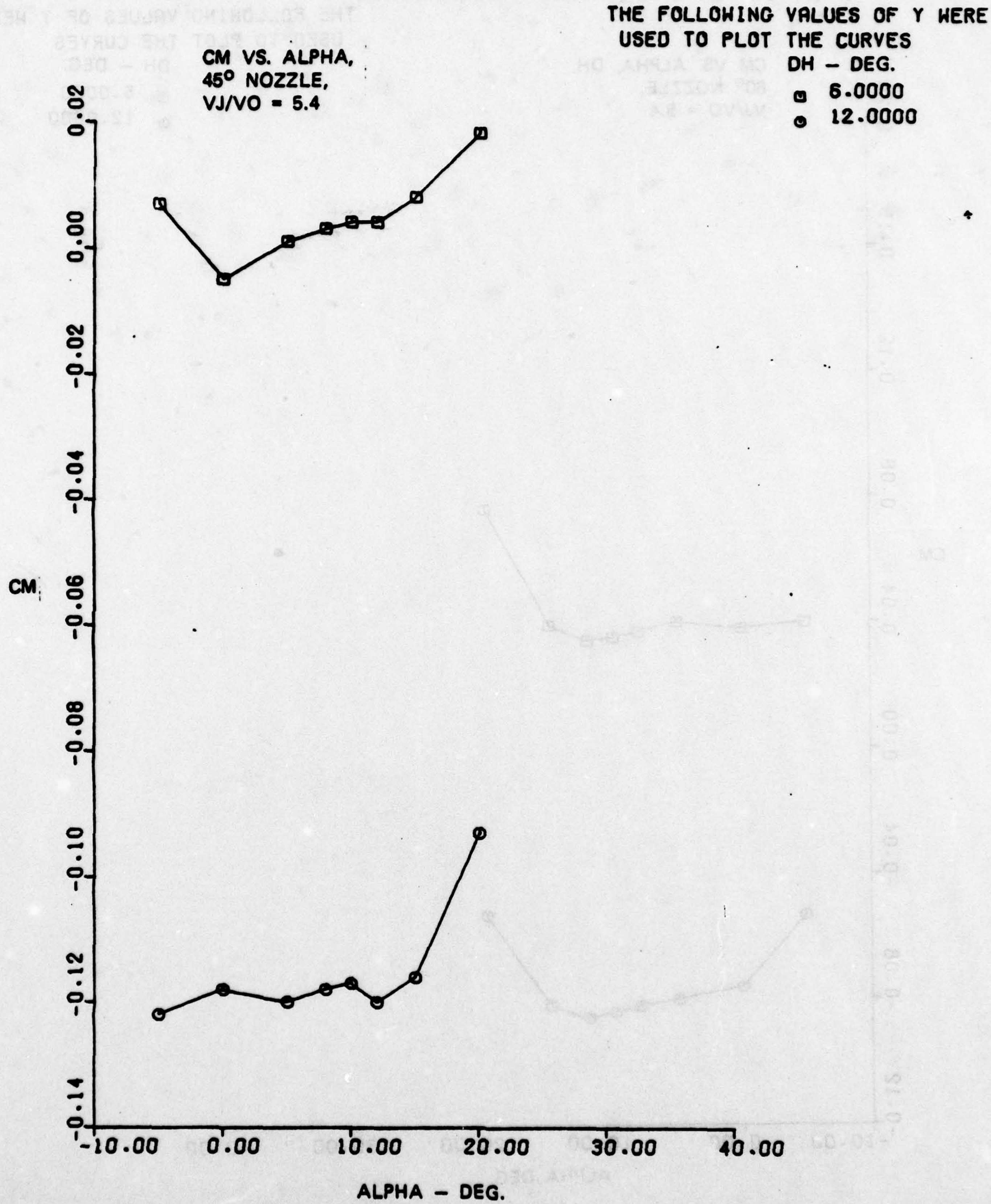

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
DH - DEG.
6.0000
12.0000
CM VS. ALPHA,
45° NOZZLE,
VJ/VO = 5.4
CM
0.02
0.00
-0.02
-0.04
-0.06
-0.08
-0.10
-0.12
-0.14
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA - DEG.

TABLE NO.58

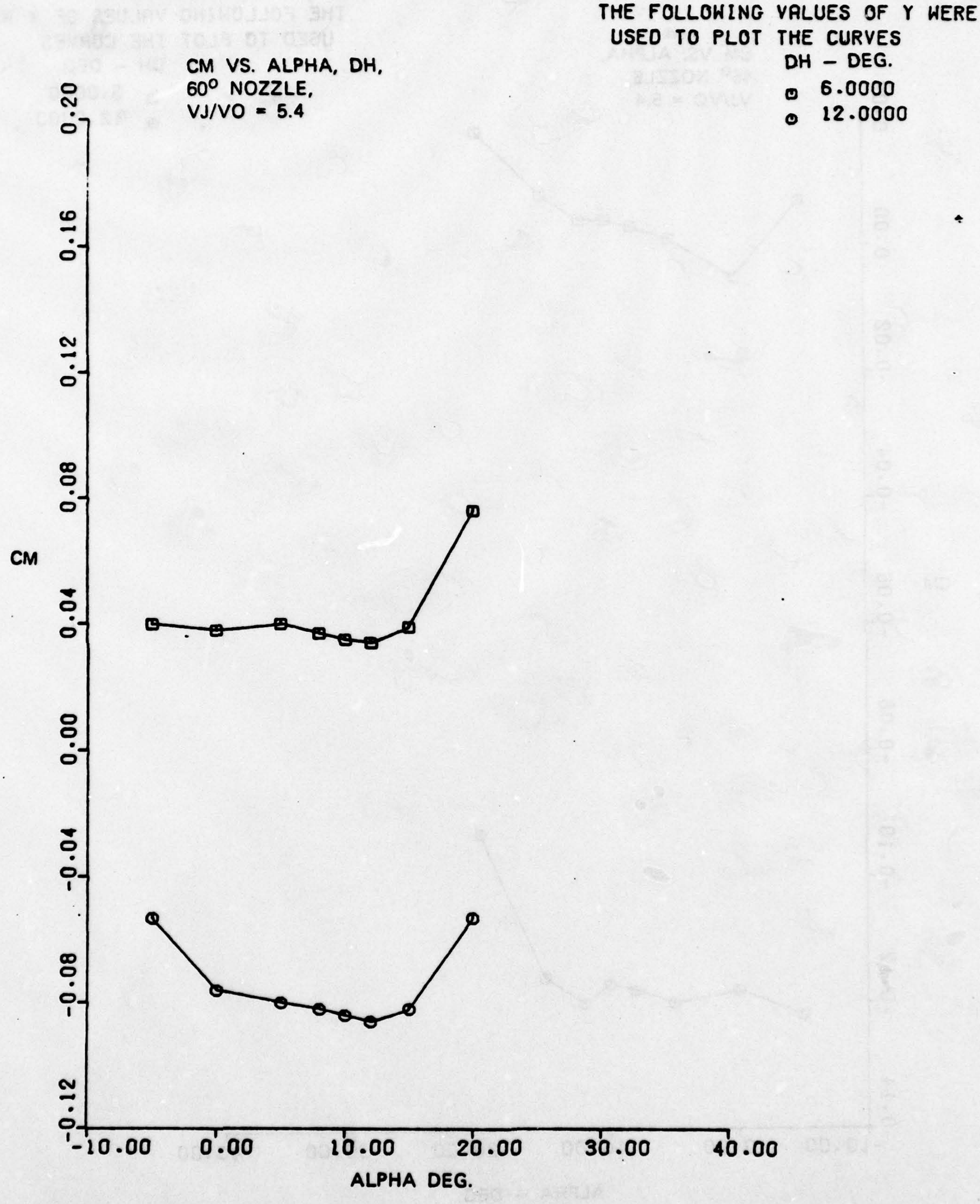

THE FOLLOWING VALUES OF Y WERE
USED TO PLOT THE CURVES
DH – DEG.
6.0000
12.0000
CM VS. ALPHA, DH,
60° NOZZLE,
VJ/VO = 5.4
CM
0.20
0.16
0.12
0.08
0.04
0.00
-0.04
-0.08
-0.12
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA DEG.

TABLE NO.59

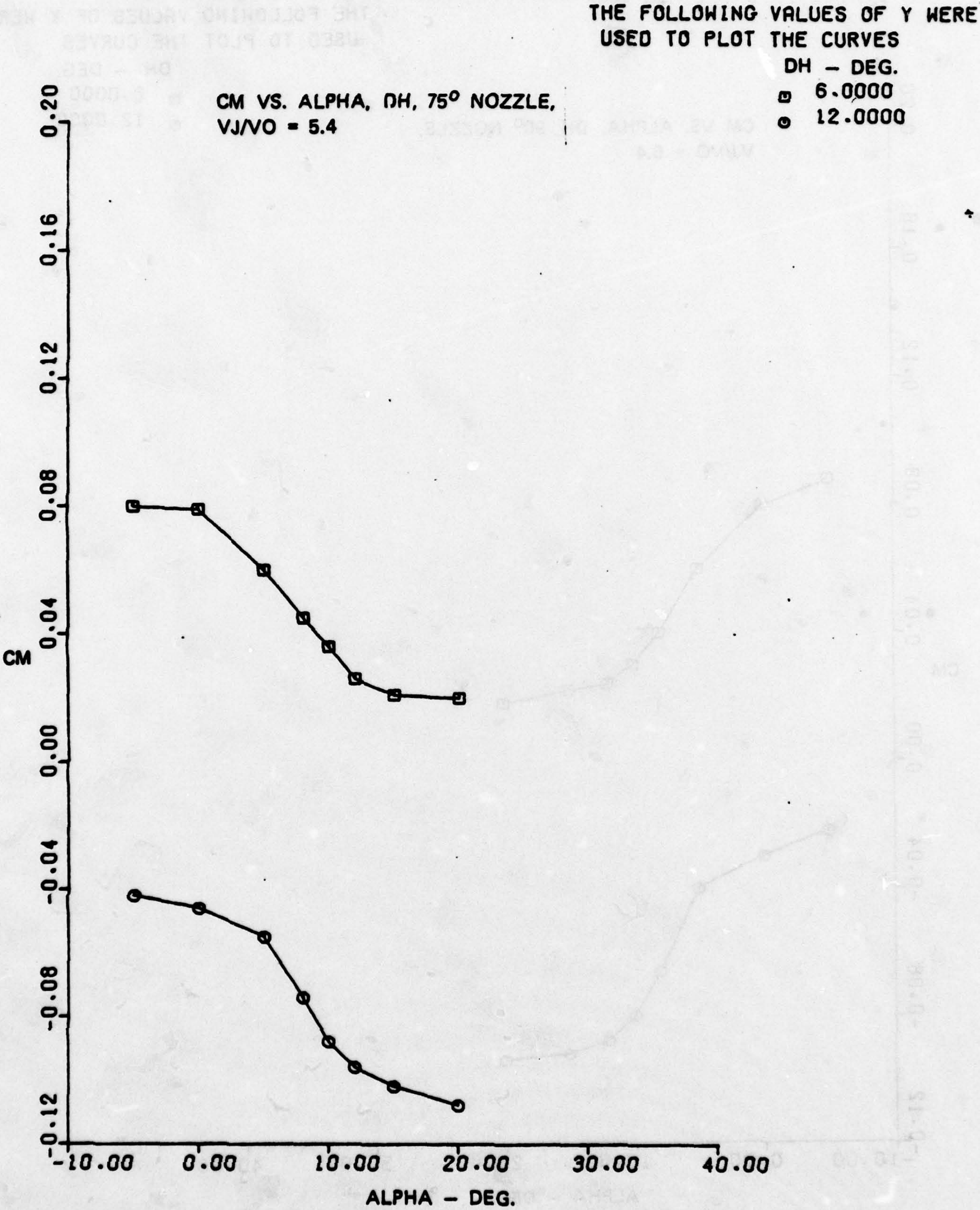

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
DH – DEG.
6.0000
12.0000
CM VS. ALPHA, DH, 75° NOZZLE, VJ/VO = 5.4
CM
0.20
0.16
0.12
0.08
0.04
0.00
-0.04
-0.08
-0.12
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA – DEG.

TABLE NO.60

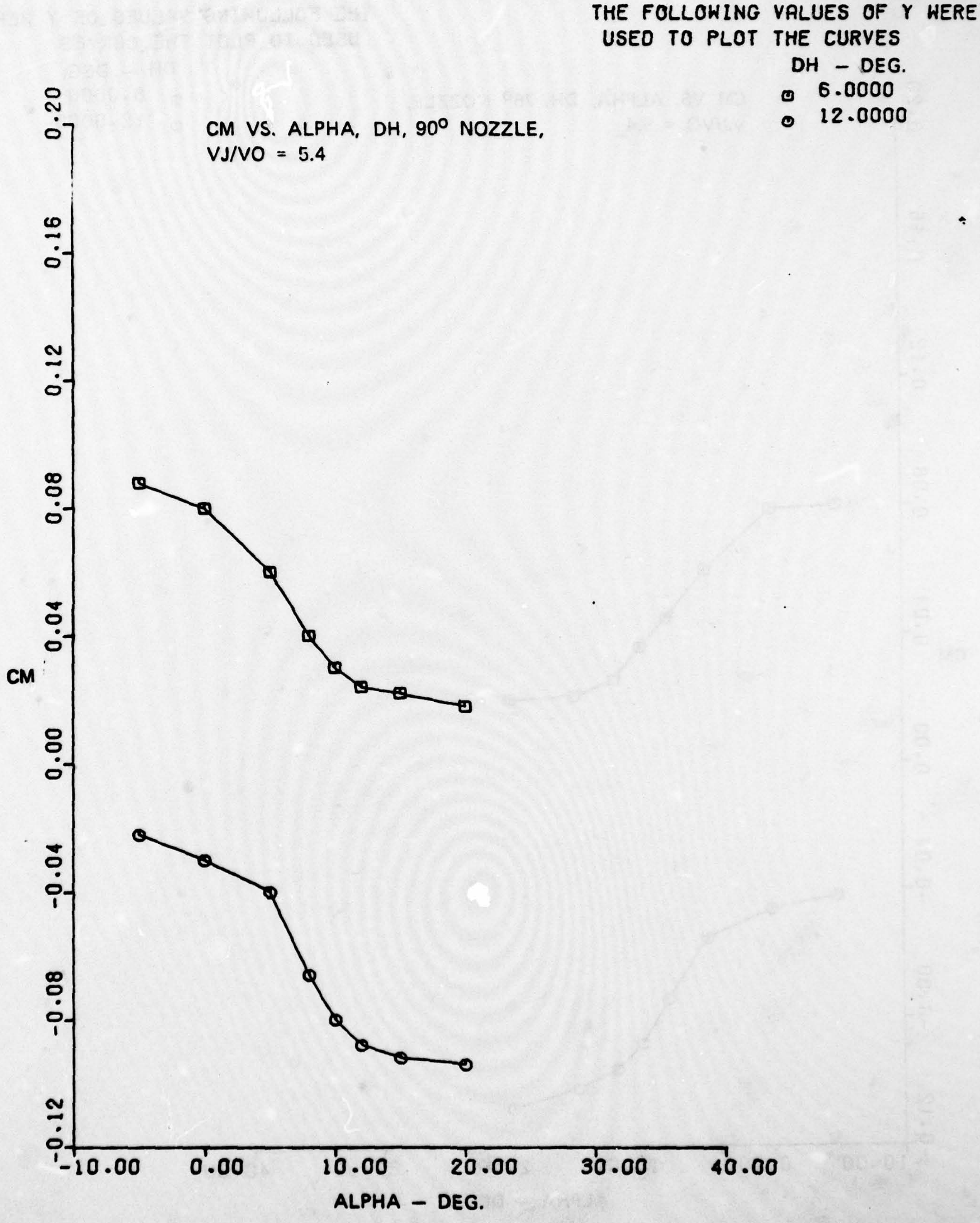

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
DH – DEG.
6.0000
12.0000
CM VS. ALPHA, DH, 90° NOZZLE,
VJ/VO = 5.4
CM
0.20
0.16
0.12
0.08
0.04
0.00
-0.04
-0.08
-0.12
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA – DEG.

TABLE NO.61

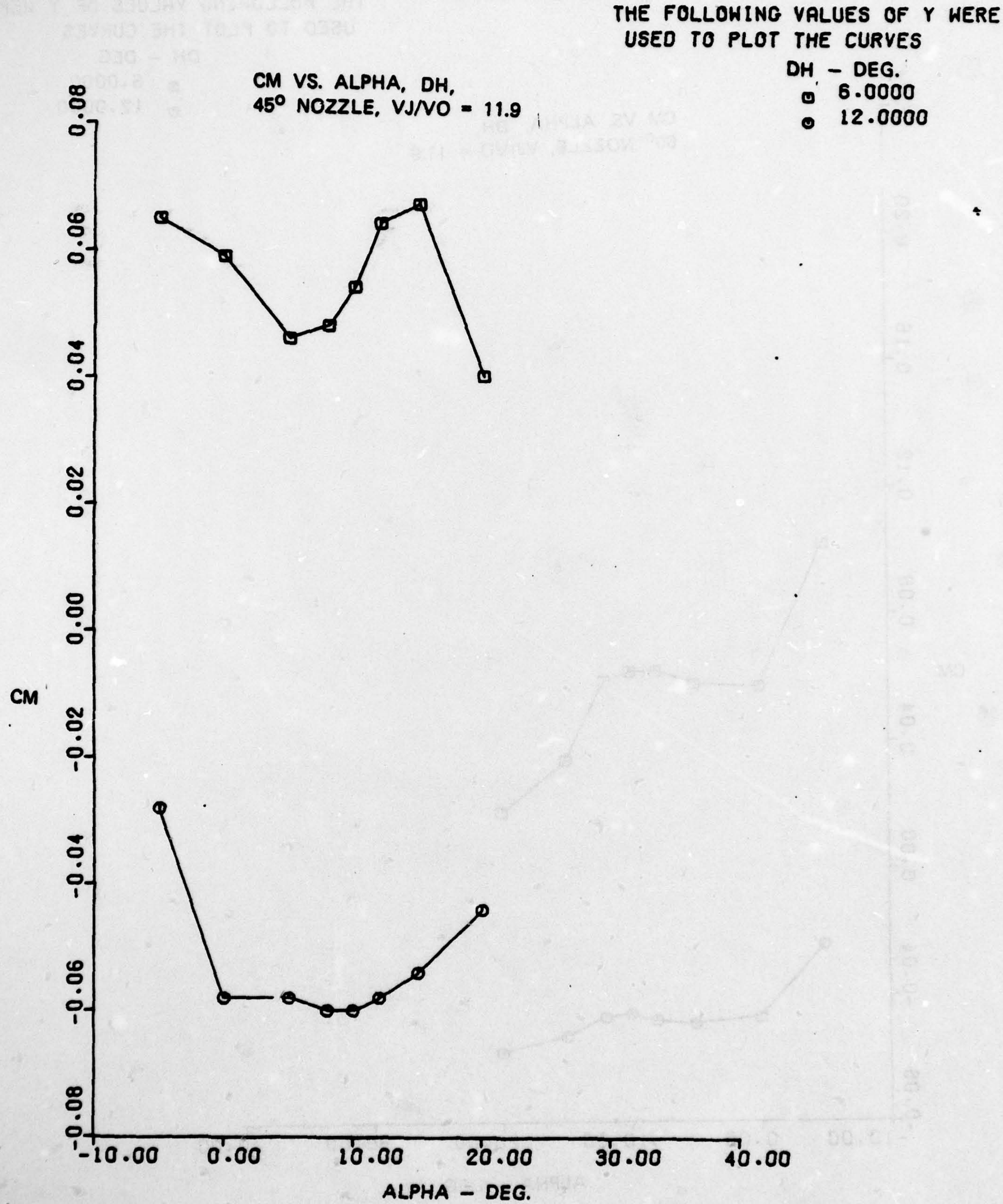

THE FOLLOWING VALUES OF Y WERE
USED TO PLOT THE CURVES
DH - DEG.
6.0000
12.0000
CM VS. ALPHA, DH,
45° NOZZLE, VJ/VO = 11.9
CM
0.08
0.06
0.04
0.02
0.00
-0.02
-0.04
-0.06
-0.08
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA - DEG.

TABLE NO.62

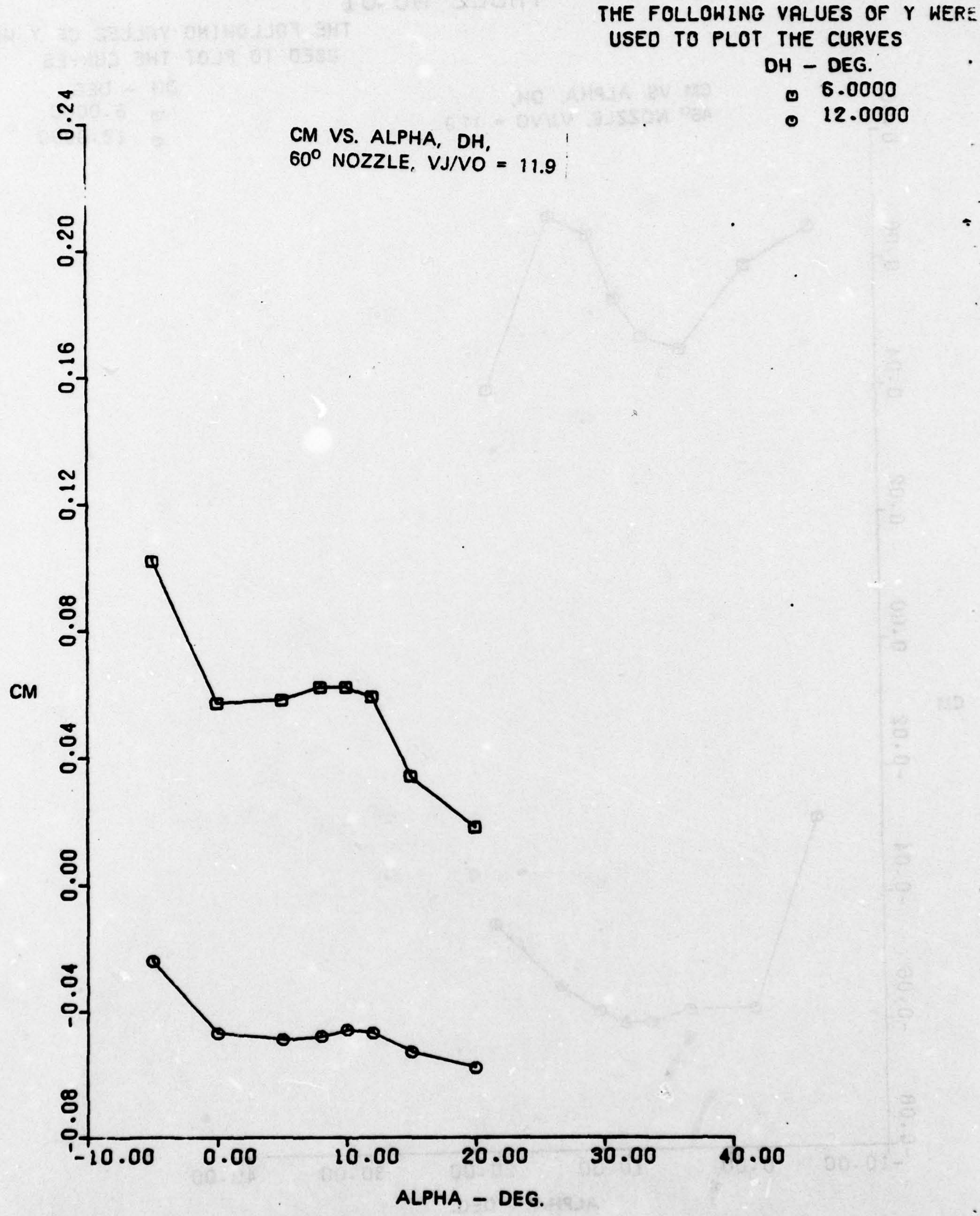

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
DH – DEG.
6.0000
12.0000
CM VS. ALPHA, DH,
60° NOZZLE, VJ/VO = 11.9
CM
0.24
0.20
0.16
0.12
0.08
0.04
0.00
-0.04
-0.08
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA – DEG.

TABLE NO.63

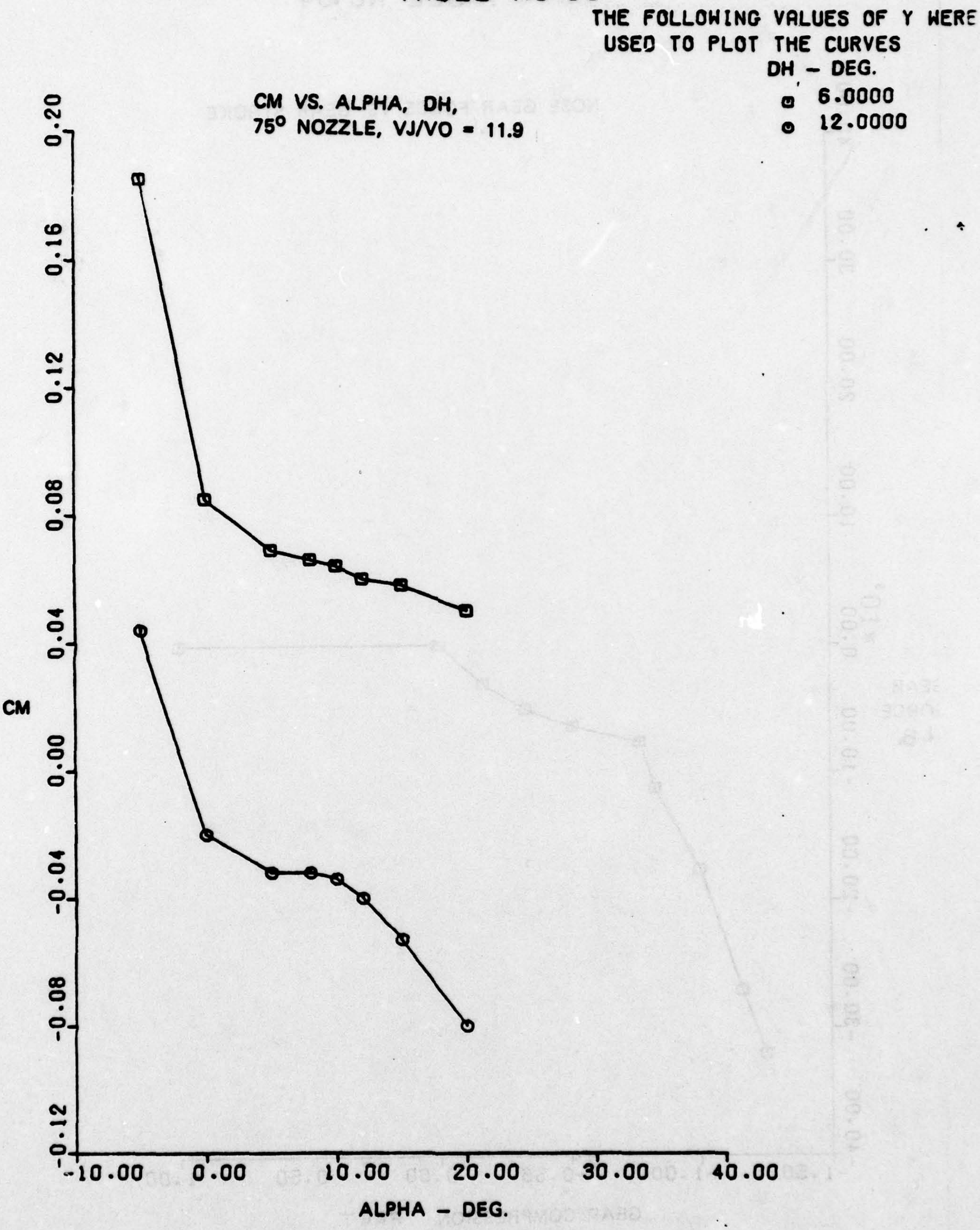

THE FOLLOWING VALUES OF Y WERE
USED TO PLOT THE CURVES
DH – DEG.
6.0000
12.0000
CM VS. ALPHA, DH,
75° NOZZLE, VJ/VO = 11.9
CM
0.20
0.16
0.12
0.08
0.04
0.00
-0.04
-0.08
-0.12
-10.00
0.00
10.00
20.00
30.00
40.00
ALPHA – DEG.

TABLE NO.64

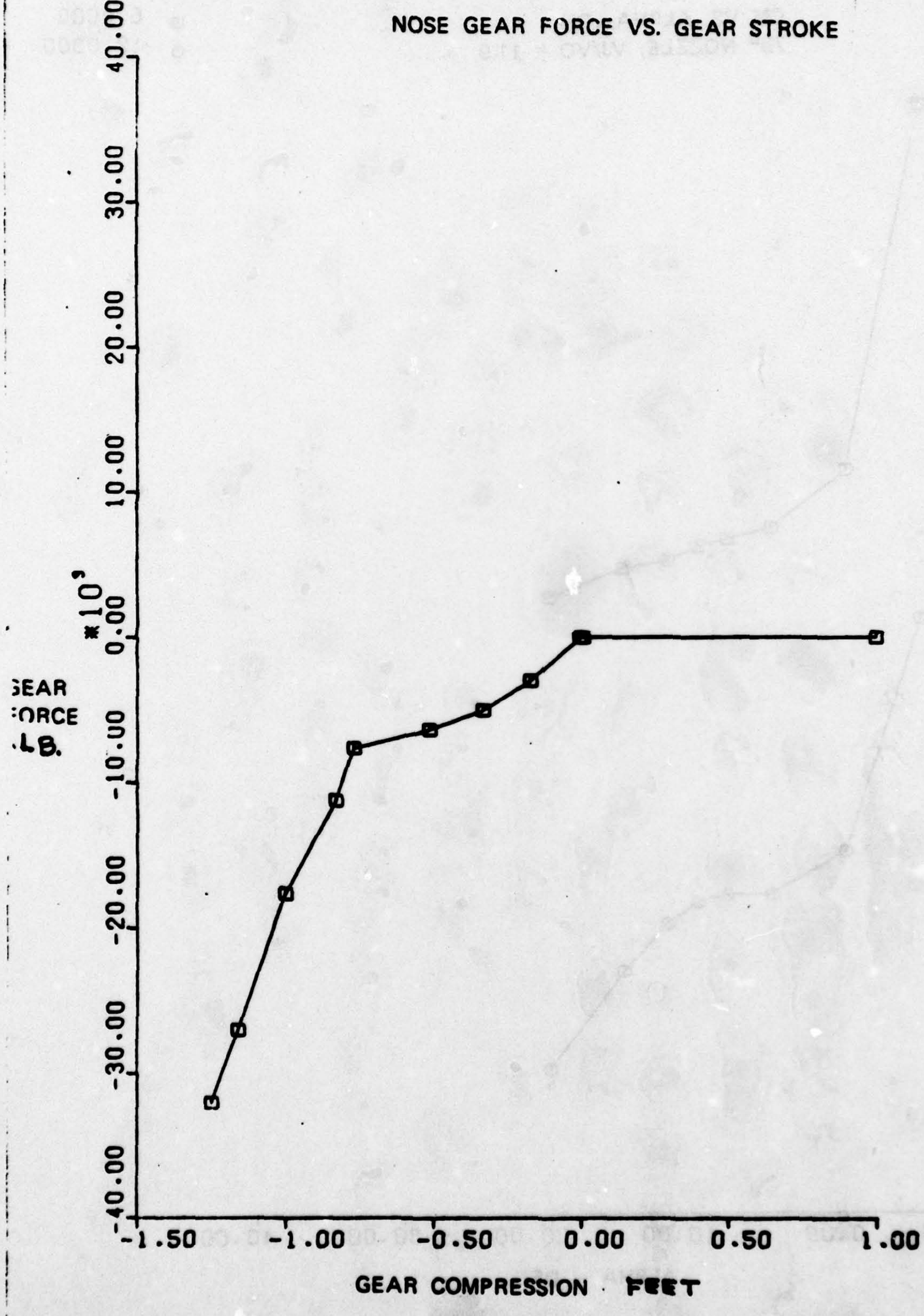

NOSE GEAR FORCE VS. GEAR STROKE
GEAR FORCE LB.
*10³
40.00
30.00
20.00
10.00
0.00
-10.00
-20.00
-30.00
-40.00
-1.50
-1.00
-0.50
0.00
0.50
1.00
GEAR COMPRESSION FEET

TABLE NO.65

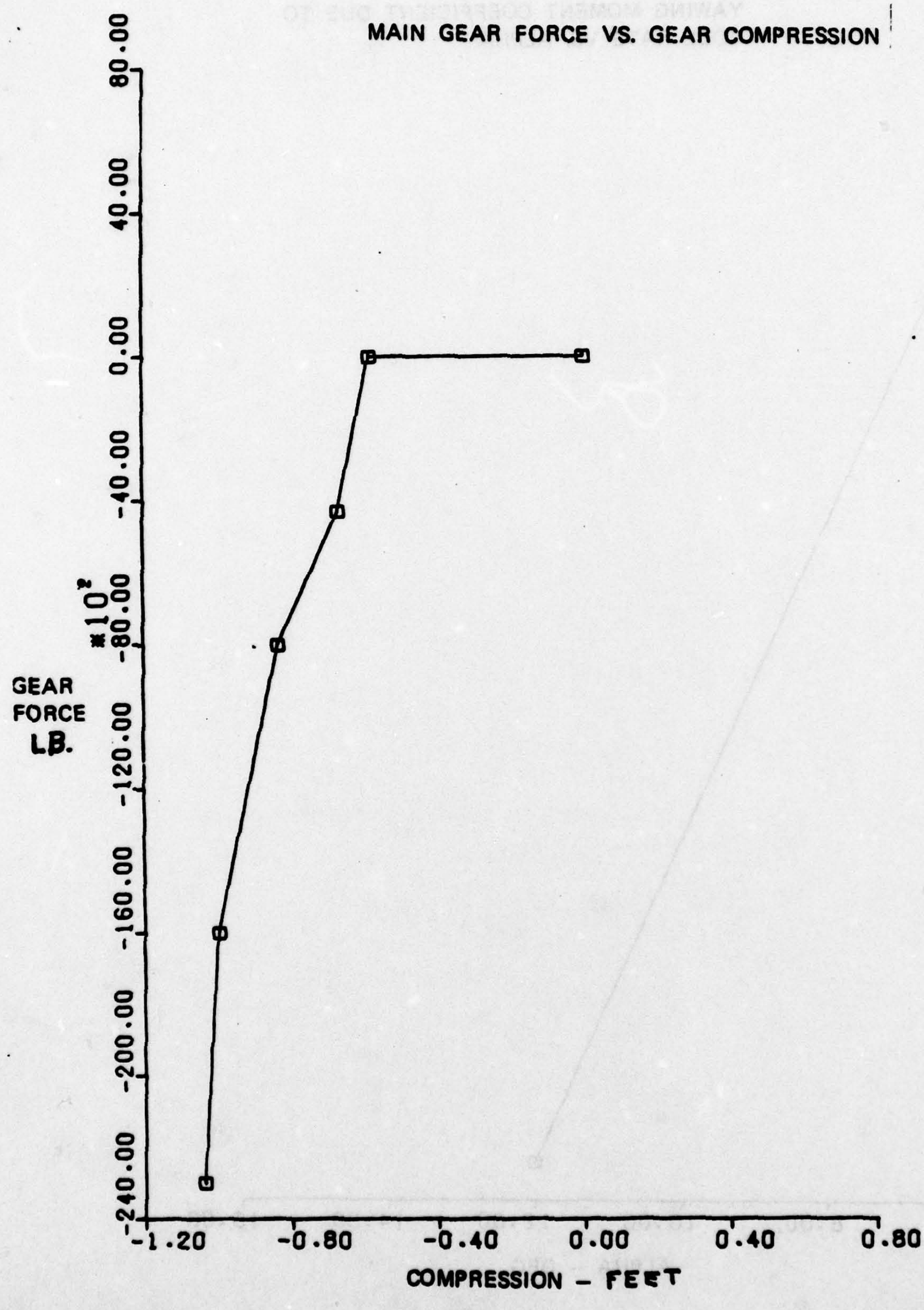

MAIN GEAR FORCE VS. GEAR COMPRESSION
GEAR FORCE LB.
*10²
80.00
40.00
0.00
-40.00
-80.00
-120.00
-160.00
-200.00
-240.00
-1.20
-0.80
-0.40
0.00
0.40
0.80
COMPRESSION – FEET

TABLE NO.66

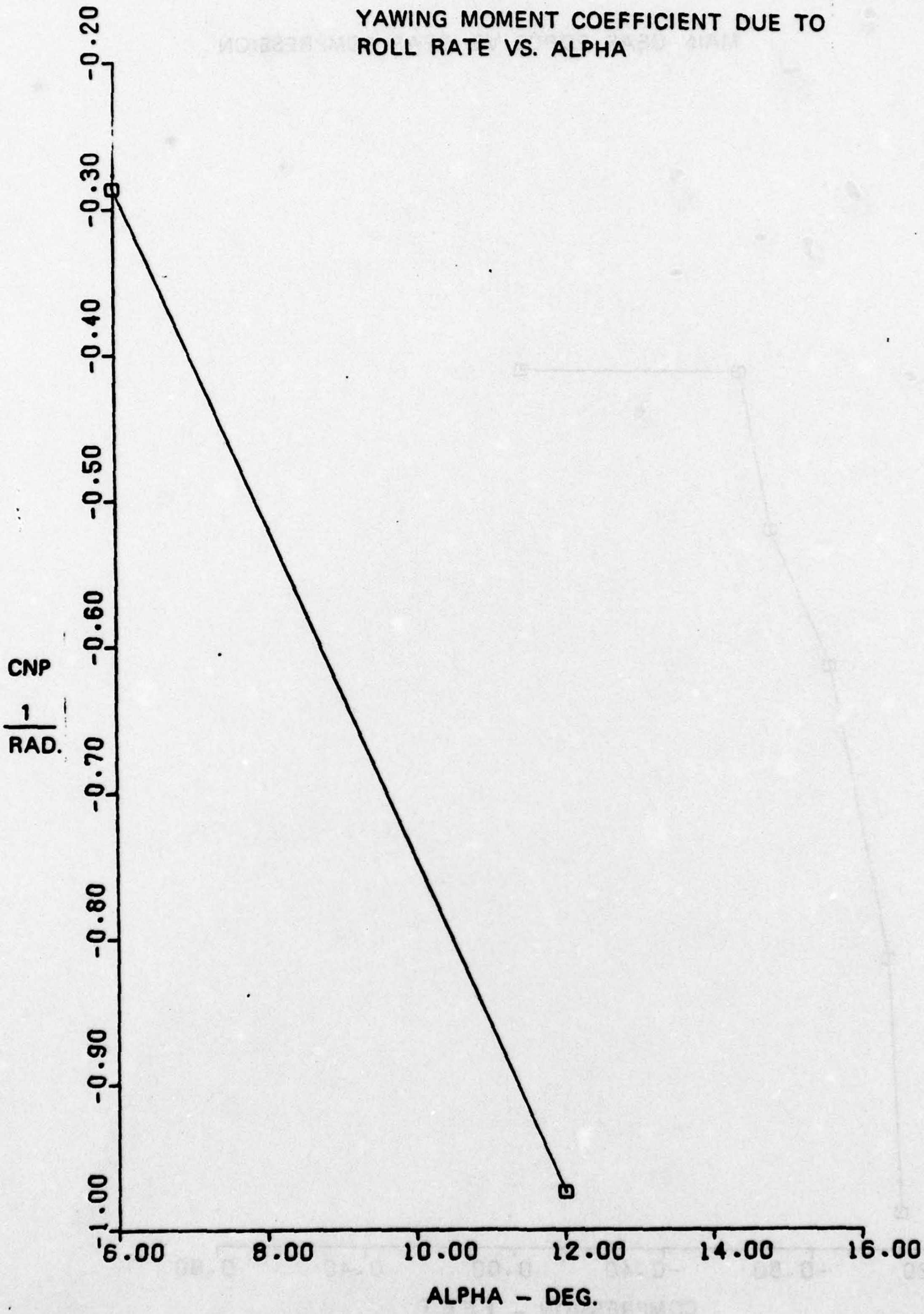

YAWING MOMENT COEFFICIENT DUE TO
ROLL RATE VS. ALPHA
CNP
1
RAD.
-0.20
-0.30
-0.40
-0.50
-0.60
-0.70
-0.80
-0.90
-1.00
6.00
8.00
10.00
12.00
14.00
16.00
ALPHA - DEG.

TABLE NO.67

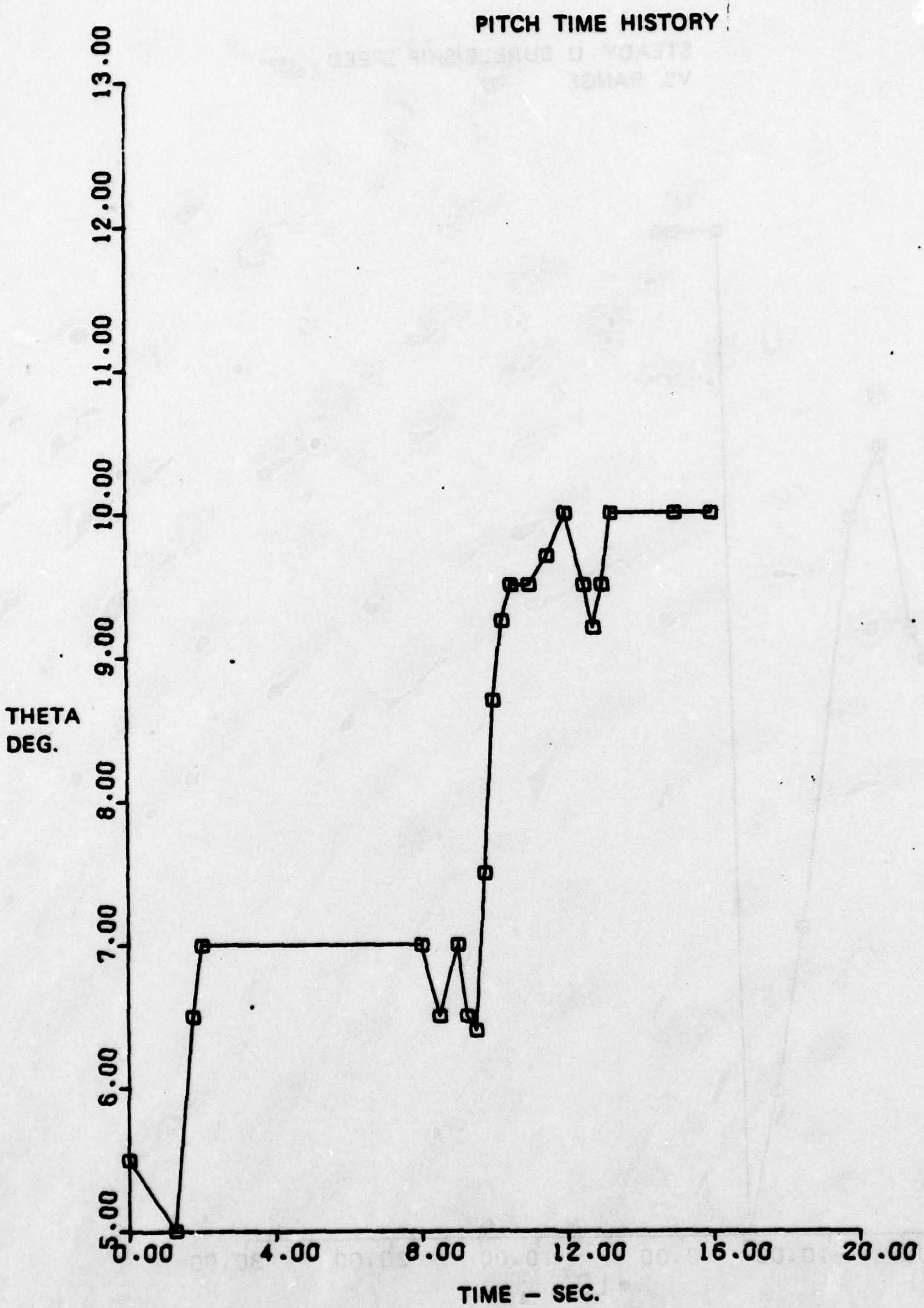

PITCH TIME HISTORY
THETA
DEG.
13.00
12.00
11.00
10.00
9.00
8.00
7.00
6.00
5.00
0.00
4.00
8.00
12.00
16.00
20.00
TIME — SEC.

TABLE NO.68

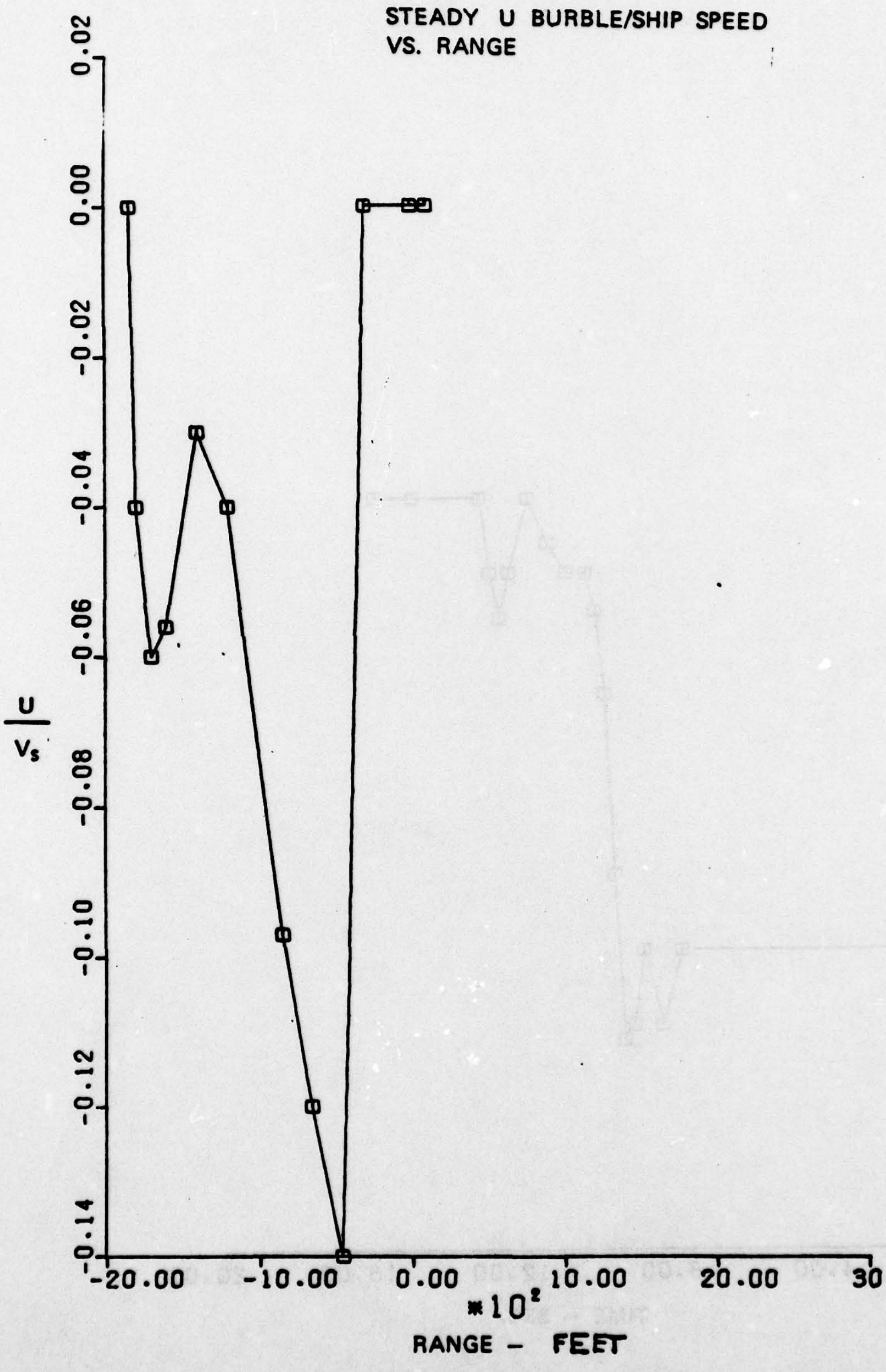

STEADY U BURBLE/SHIP SPEED
VS. RANGE
0.02
0.00
-0.02
-0.04
-0.06
-0.08
-0.10
-0.12
-0.14
U / V_s
-20.00
-10.00
0.00
10.00
20.00
30.00
*10²
RANGE - FEET

TABLE NO.69

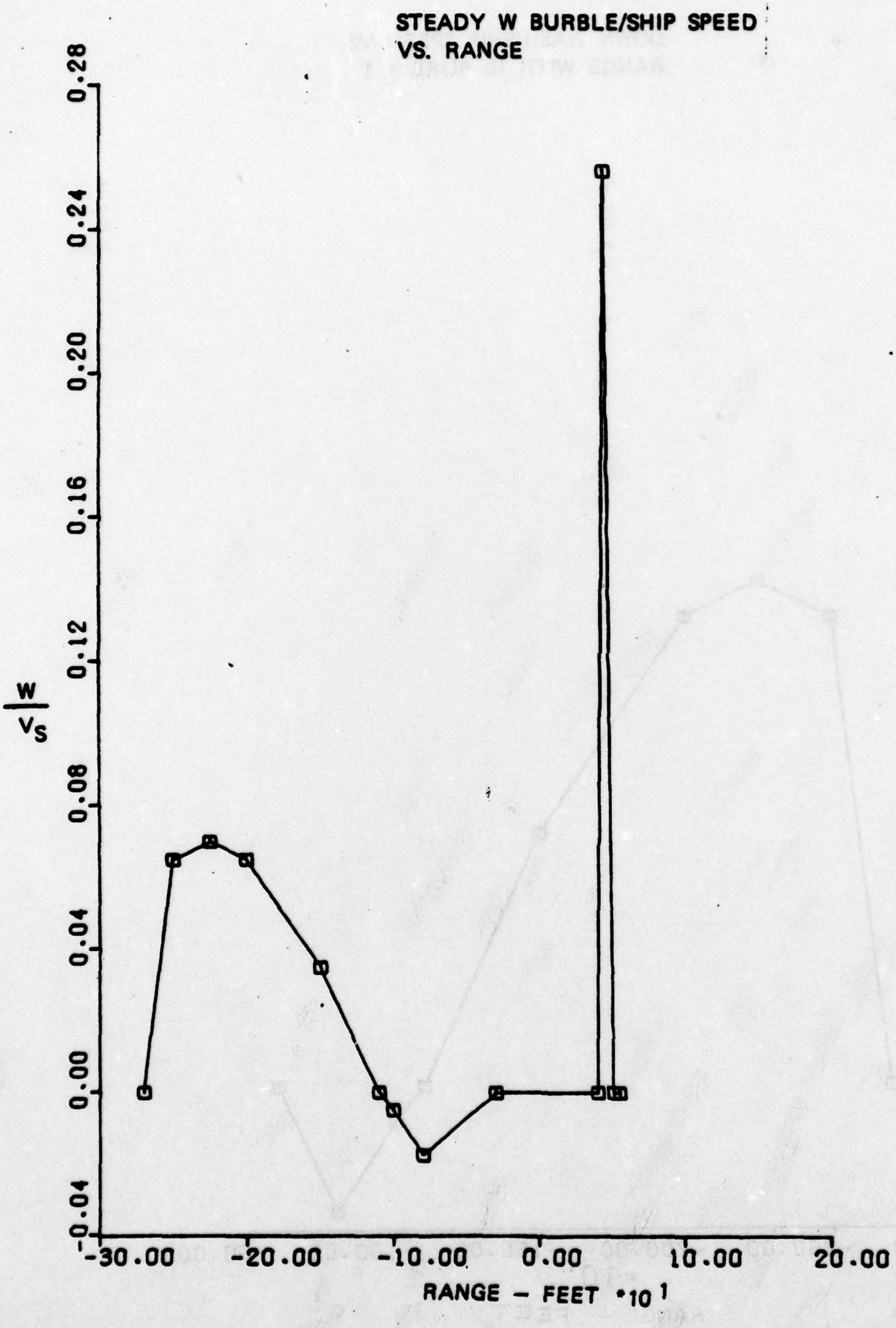

STEADY W BURBLE/SHIP SPEED
VS. RANGE
0.28
0.24
0.20
0.16
0.12
0.08
0.04
0.00
-0.04
W/Vs
-30.00
-20.00
-10.00
0.00
10.00
20.00
RANGE – FEET *10 1

TABLE NO.70

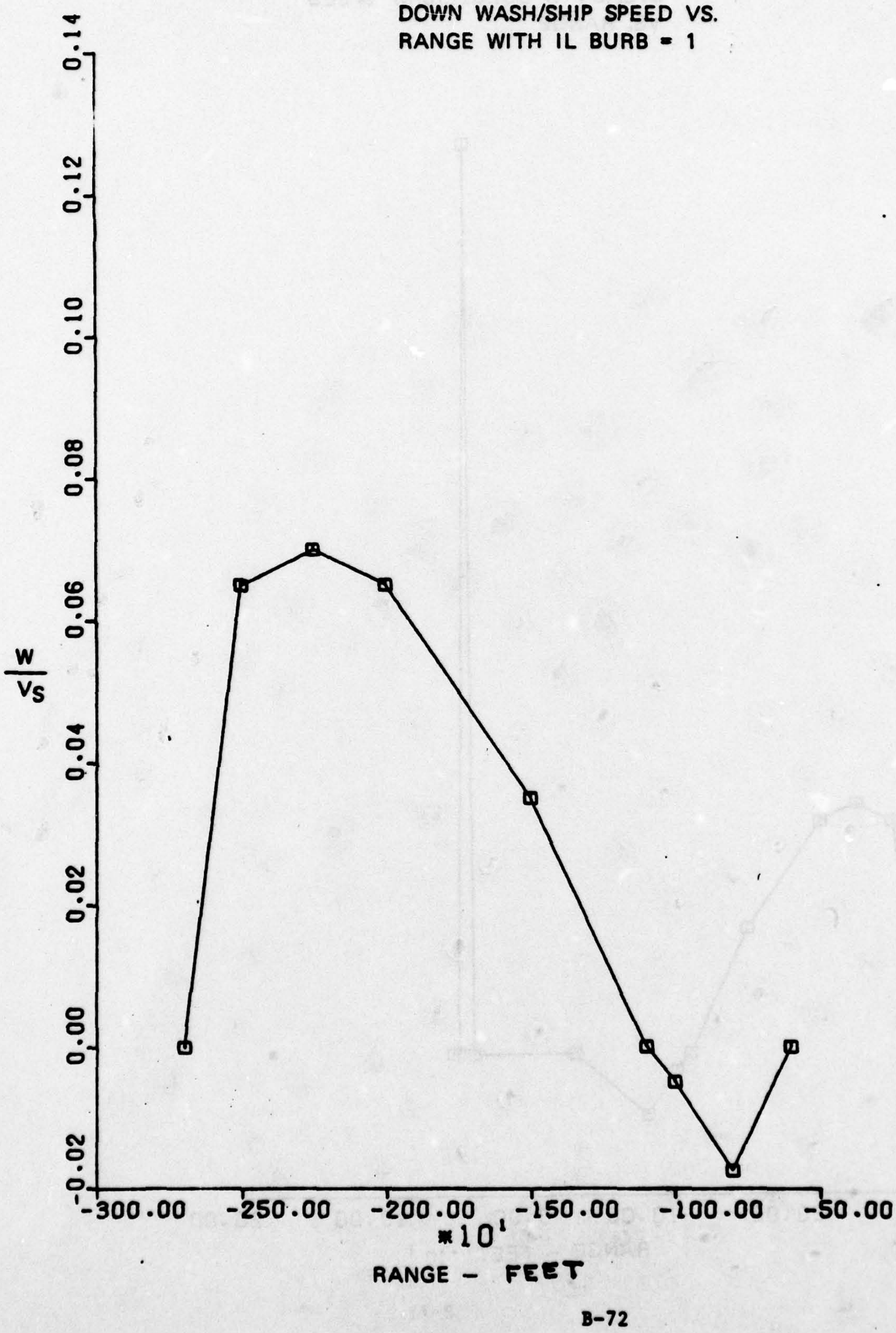

DOWN WASH/SHIP SPEED VS.
RANGE WITH IL BURB = 1
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
-0.02
W/VS
-300.00
-250.00
-200.00
-150.00
-100.00
-50.00
*10^1
RANGE - FEET

TABLE NO.71

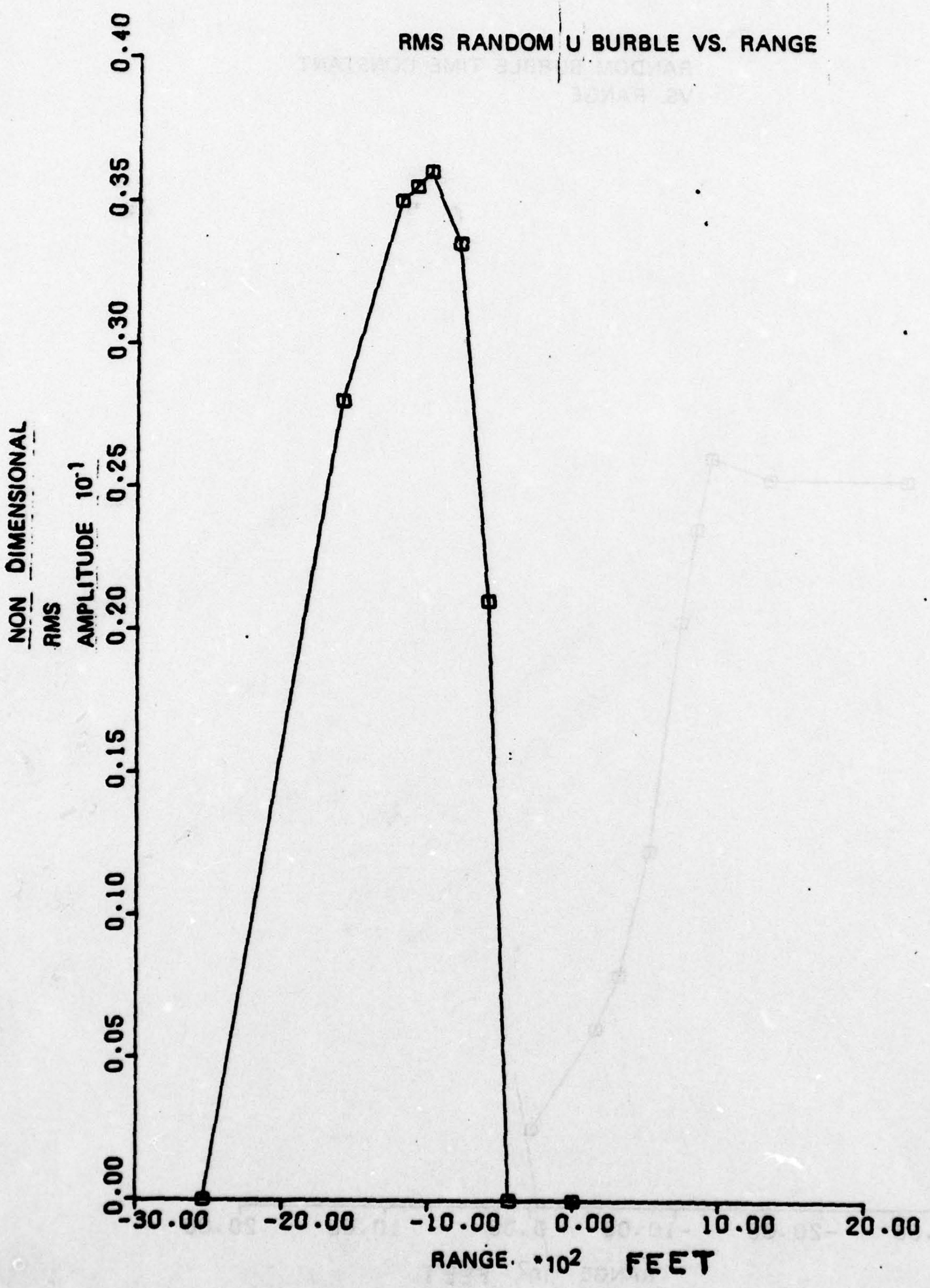

RMS RANDOM U BURBLE VS. RANGE
NON DIMENSIONAL
RMS
AMPLITUDE 10^-1
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
-30.00
-20.00
-10.00
0.00
10.00
20.00
RANGE *10^2 FEET

TABLE NO.72

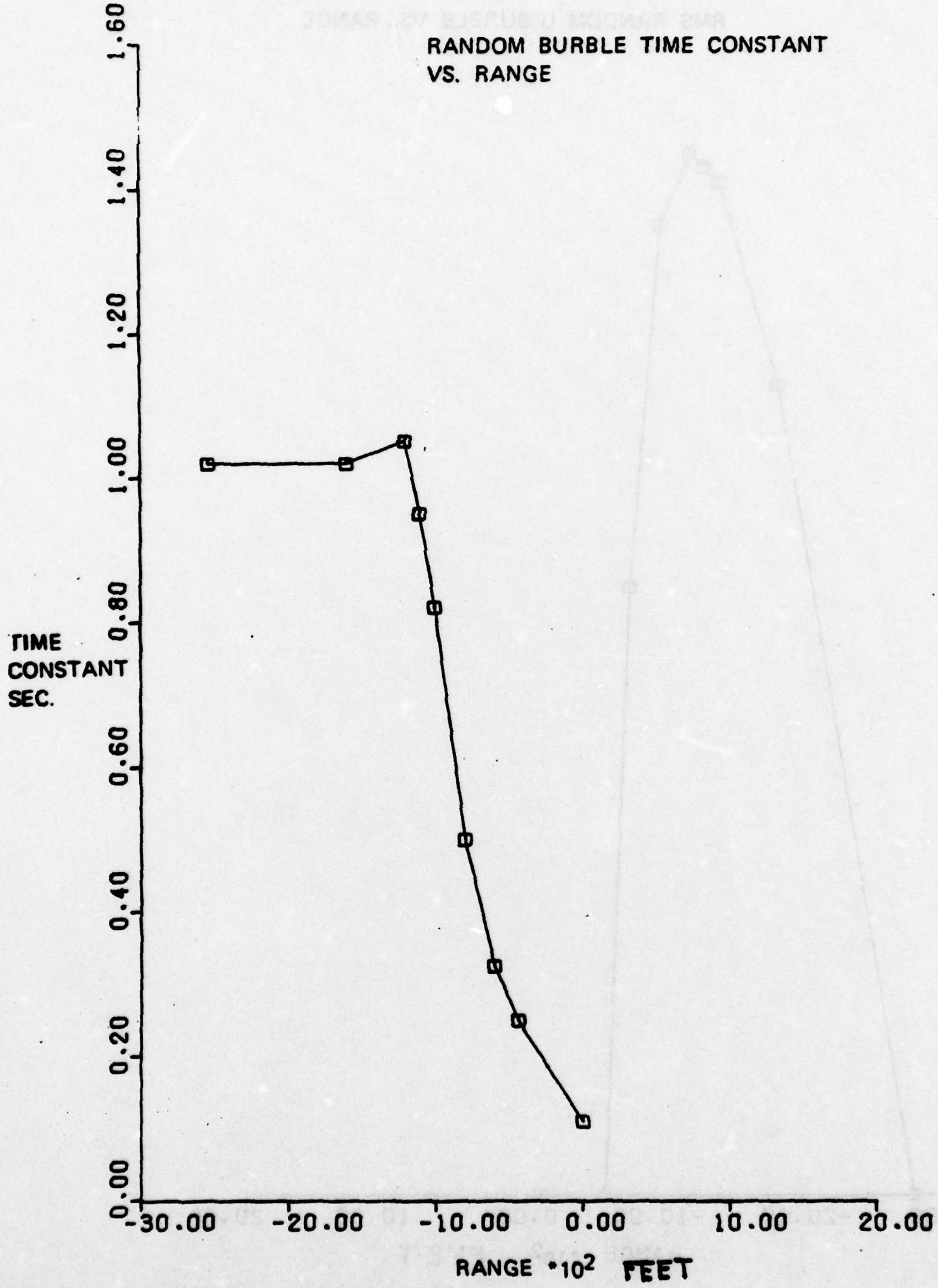

RANDOM BURBLE TIME CONSTANT
VS. RANGE
TIME
CONSTANT
SEC.
0.00
0.20
0.40
0.60
0.80
1.00
1.20
1.40
1.60
-30.00
-20.00
-10.00
0.00
10.00
20.00
RANGE *10^2 FEET

TABLE NO.73

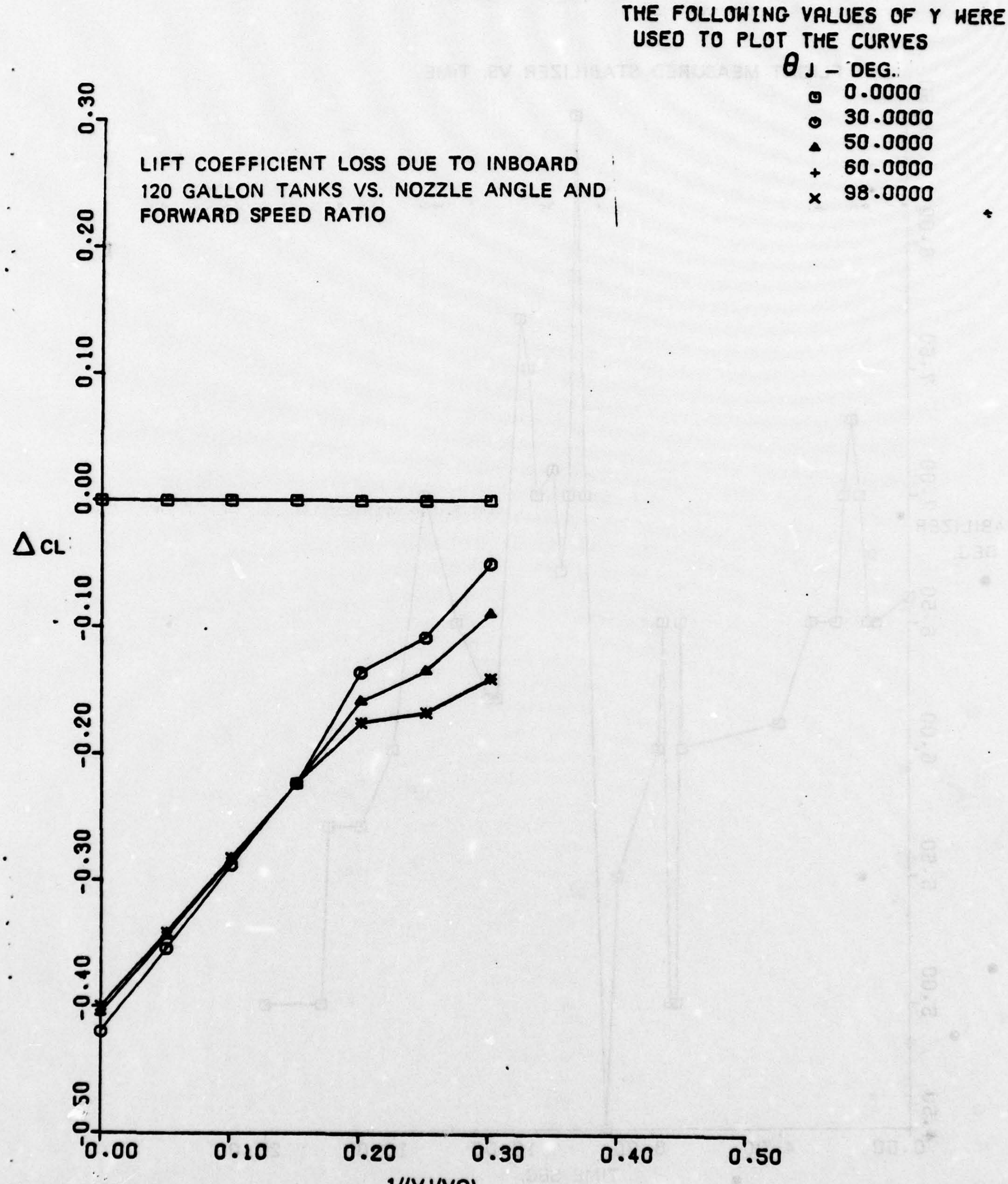

THE FOLLOWING VALUES OF Y WERE USED TO PLOT THE CURVES
θ J - DEG.
0.0000
30.0000
50.0000
60.0000
98.0000
LIFT COEFFICIENT LOSS DUE TO INBOARD 120 GALLON TANKS VS. NOZZLE ANGLE AND FORWARD SPEED RATIO
0.30
0.20
0.10
0.00
-0.10
-0.20
-0.30
-0.40
-0.50
ΔCL
0.00
0.10
0.20
0.30
0.40
0.50
1/(VJ/VO)

TABLE NO.74

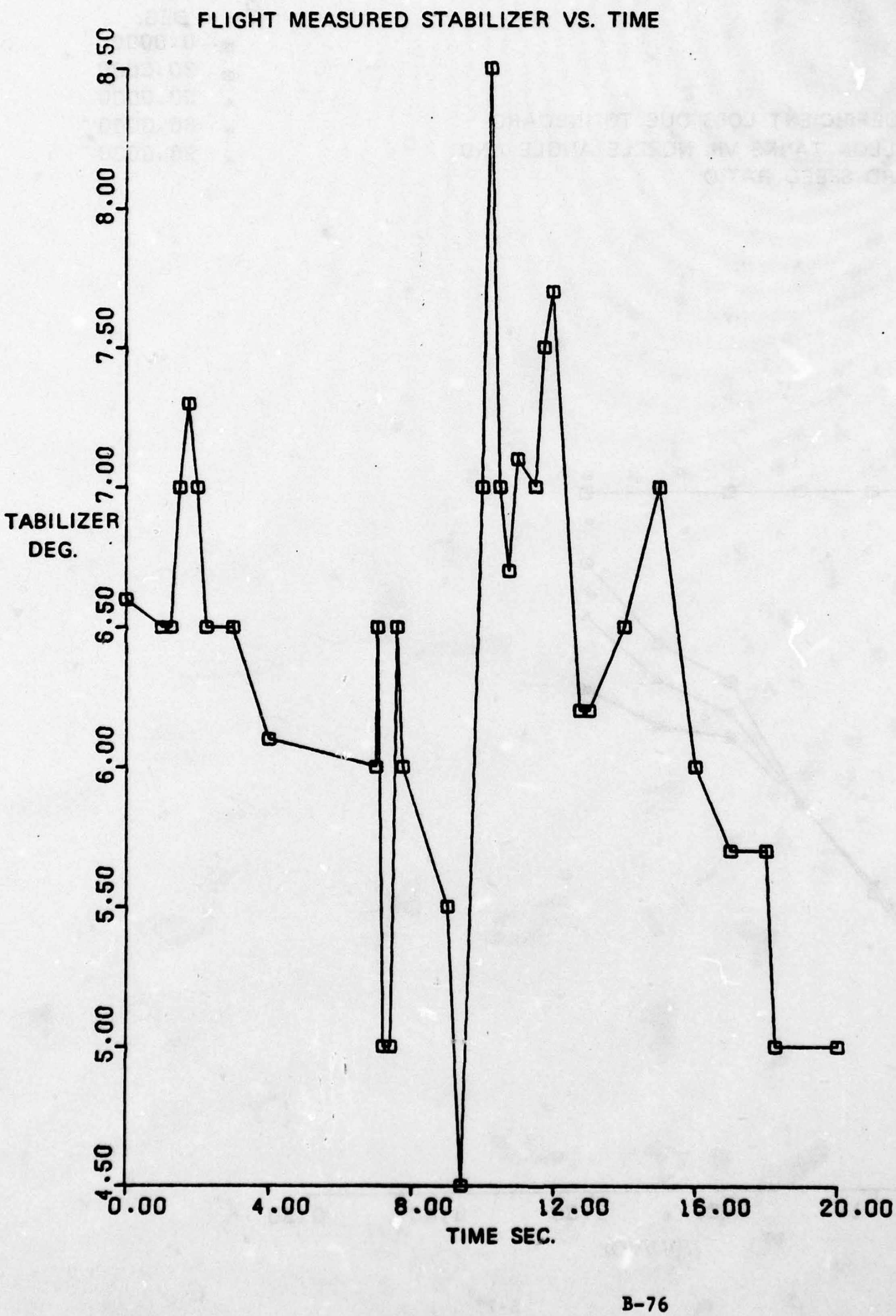
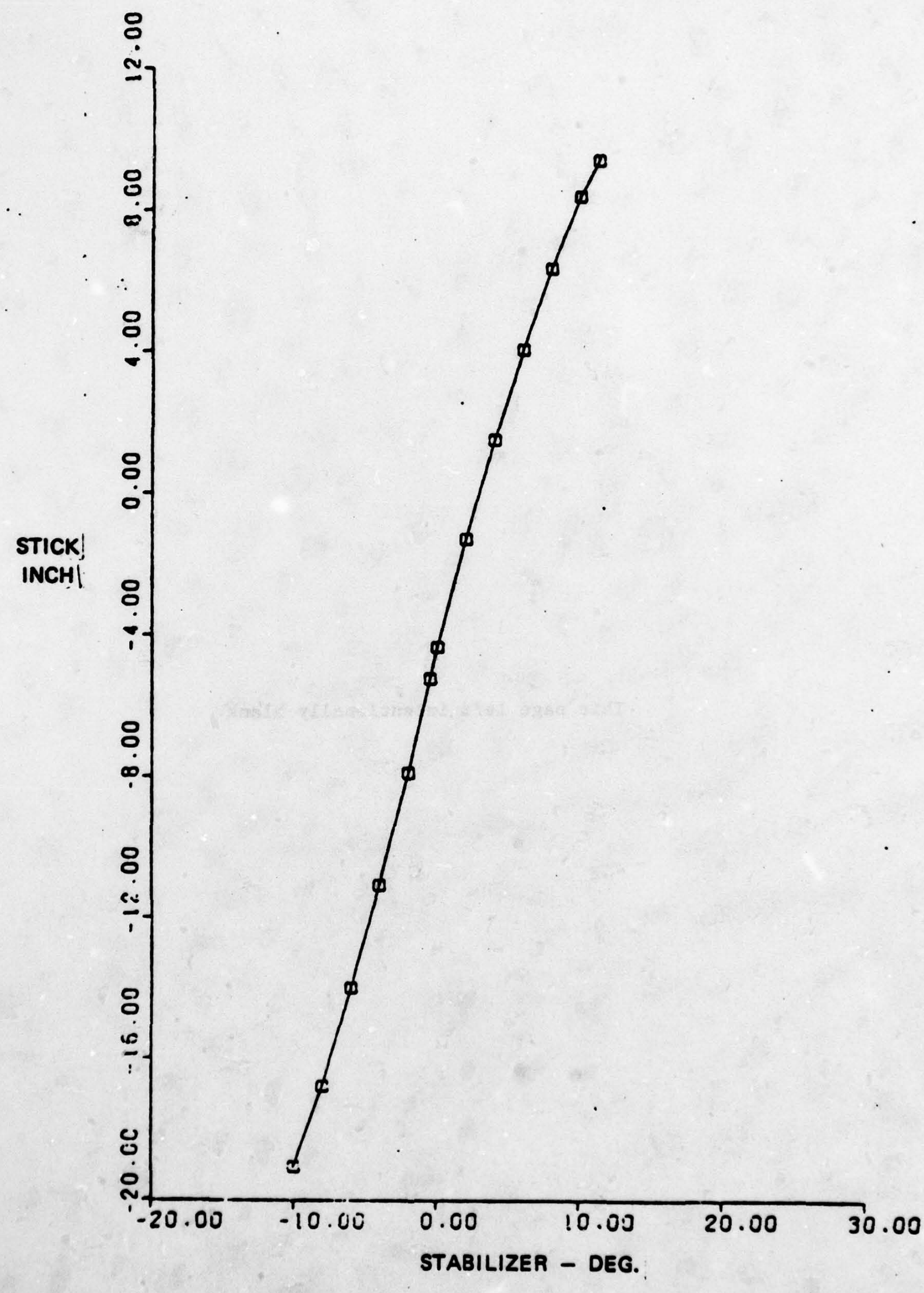

FLIGHT MEASURED STABILIZER VS. TIME
TABILIZER DEG.
8.50
8.00
7.50
7.00
6.50
6.00
5.50
5.00
4.50
0.00
4.00
8.00
12.00
16.00
20.00
TIME SEC.

TABLE NO.75

LONGITUDINAL STICK VS. STABILIZER



This page left intentionally blank.

APPENDIX C

USER'S GUIDE TO THE VSTOL MODEL PROGRAM

This page left intentionally blank.

APPENDIX C
USER'S GUIDE TO THE VSTOL MODEL PROGRAM

The following description explains the procedures for loading data required for the simulation. The principal program variables reside in labeled common blocks to facilitate access between various subroutines. Initial values of common variables may be defined by using the block data subroutine. This method is used tc define basic or reference values of program variables. In addition local variables may be defined by using data statements in individual subroutines. Figure 22 shows the subroutine arrangement.

Common variables may be changed from run to run by specifying appropriate data cards located after the control card and program deck.

The common variables are divided into seven main common groups. These are as follows:

A(1-500)	Basic real program variables
IA(1-200)	Basic integer program variables
B(1-200)	User floating point variables
IB(1-50)	User fixed point variables
GA(1-100)	User control system gain parameters
SCALE(1-28)	Plotting program scale values
NPLOT(1-14)	Plot selection switch values

The format for data input is specified below:

CARD NO.	COLUMN	FORMAT	QUANTITY
1	1-2	(12)	NRUN - number of runs 1 = single run >1 = multiple runs 0 = terminate program
2	2-5	(1X, A4)	Name - input option switch = data for common variable input
3-11	1	A1	Blank
	2	I1	Common variable type indicator 1 = A common 2 = IA common 3 = GA common 4 = Scale common 5 = NPLOT(I) 6 = CHTSP 7 = B common 8 = IB common

CARD NO.	COLUMN	FORMAT	QUANTITY
3-11	8-10	I3	Common location = 1 → 500
3-11	11-20	G10.0	Value of common variable
3-11	21-80	15A4	Variable description title
12	1	A1	/ - A slash in column 1 indicates the end of common input data
13	1-5	1X,A4	Name = UDAT, indicates table data input
14-32	The format for loading table lookup data is described seperately in Appendix D. A blank card at the end of the last set of data causes the computer to stop reading table data.		
33	2-5	A4	Name = OPRN, indicates the operation of the simulation run.
33	11-19	G9.0	RTIME - specifies simulation run time in seconds
34	1-6	3I2	I1READ, I2READ, ISCALE I1READ = 0 - continue I1READ = 1 - read IPLOT(I) I2READ = 0 - continue I2READ = 1 - read IPRNT(I) ISCALE = 0 - use SCALE(I) values for data plots ISCALE = 1 - calculate plot scales
35-36	1-80	8I10	IPLOT(I), I = 1, 14 or IPRNT(I), I = 1, 11 specifies print and plot variables
37	1-2	I2	NRUN - specify number of repetitions for next run, terminate program if NRUN = 0.
38	2-5	A4	Name = END terminates program

SAMPLE DATA INPUT

```
1
DATA
1        238   64.5          A(238)= HIC ,INITIAL AIRCRAFT ALTITUDE
2        110            1    IA(110)= INDEXT=1,  TRIM INDEX
3        86    400.          GA(86)= XCG2,Hands off Takeoff Roll Parameter
4        13    25.           SCALE(13)= maximum value OF SEVENTH PLOT VARIABLE
5        1              0    NPLOT(1)=0 DELETE FIRST PLOT
6        1     .7877         CHTSP = PLOT TIME SCALE, INCH/SECOND
7        100   60.           B(100)= HIS = DECK PITCH HEIGHT
8        29             1    IB(29)=1 ,IBRAK = BRAKE APPLICATION FLAG
4        14    -25.          SCALE(14)= MINIMUM VALUE OF SEVENTH PLOT VARIABLE
/
UDAT
            10701222         LIFT DUE TO SIDE SLIP
 0.      0.      10.         12.      14.      16.      18.      20.                        1
         0.      0.          -.005    -.015    -.022    -.035    -.042                      2
            21301111          STABILIZER VS STICK
 0.        -7.5    -6.6      -5.5     -4.35    -3.1     -2.05    -1.7     -.5      .6       1
           -10.    -8.       -6.      -4.      -2.      -.5      0.       2.       4.       2
            1.6     2.5      3.2      3.7                                                   3
            6.      8.       10.      11.5                                                  4
            30503111         CY VS BETA,ALPHA, THETJ =60. ,VJ/VO=11.9
60.        -20.    -10.      0.       10.      20.                                          1
 0.         .3      .17      0.       -.2      -.4                                          2
 5.         .36     .18      0.       -.18     -.36                                         3
10.         .36     .18      0.       -.16     -.32                                         4
            40503111         CY VS BETA,ALPHA, THETJ=90. ,VJ/VO=11.9
90.        -20.    -10.      0.       10.      20.                                          1
 0.         .4      .23      0.       -.18     -.35                                         2
```

PUNCH CARD TRANSCRIPT 4ND-NADC-5230/2 (REV. 4-68) C-5

PLATE NO.

NADC-77024-30

30	10.	.38	.17	0.	-.18	-.31		3
31	20.	.27	.135	0.	-.135	-.27		4
32								
33	OPRN	15.						
34	01 0001							
35	12001	122009	122011	12016	12019	11002	11001	11003
36	51070	51083	51080	52140	11025	52142		
37	O							
38	END							
10	7/8/9							
11	6/7/8/9							

C-6

PUNCH CARD TRANSCRIPT 4ND-NADC-5230/2 (REV. 4-68)

PLATE NO. 10420

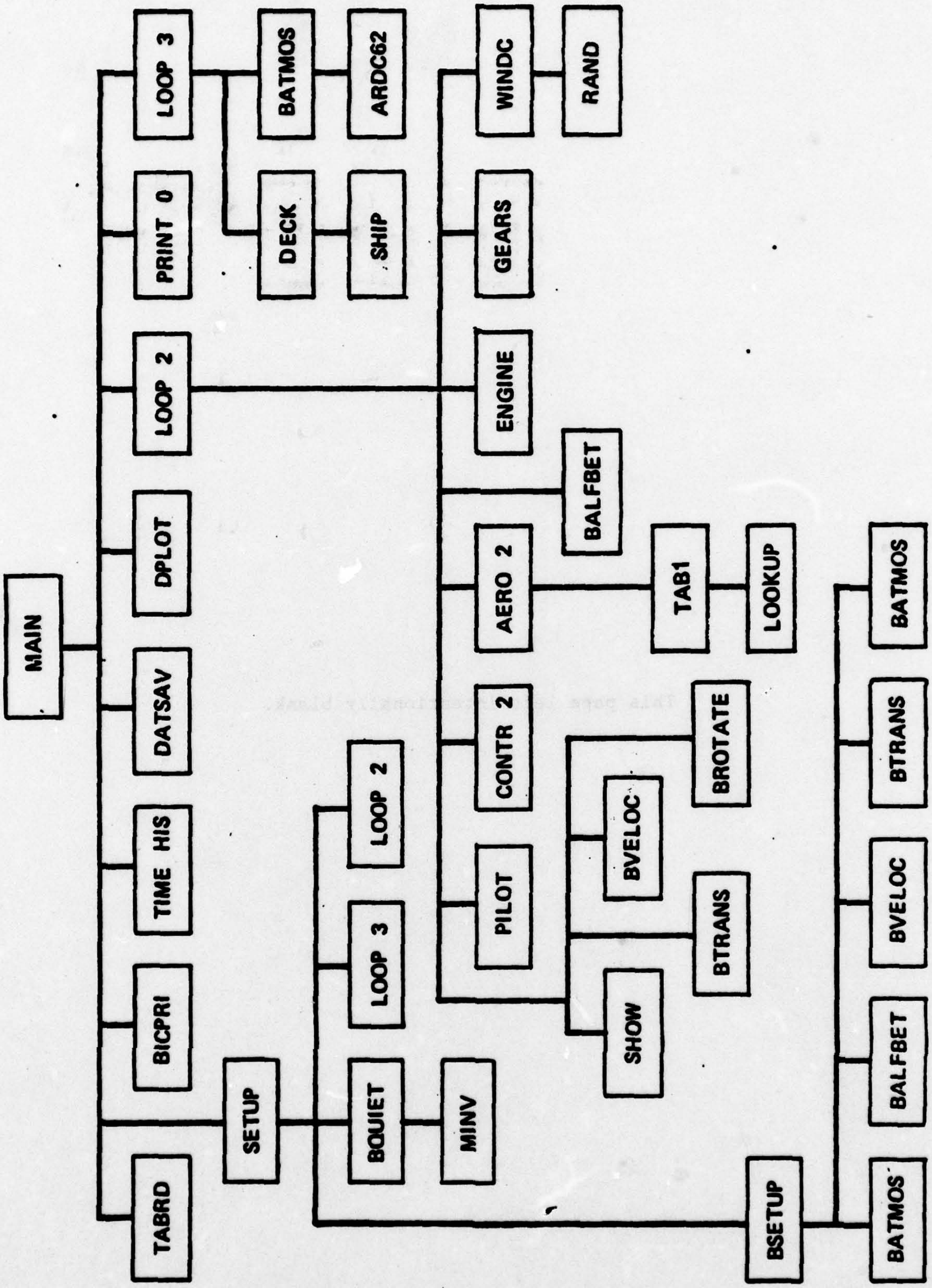


FIGURE 22. SUBROUTINE CALLING SEQUENCE

This page left intentionally blank.

APPENDIX D

USER'S GUIDE FOR LINEAR TABLE DATA LOOKUP ROUTINE

This page left intentionally blank.

APPENDIX D
USER'S GUIDE FOR LINEAR TABLE DATA LOOKUP ROUTINE

Three FORTRAN subroutines, TABRD, LOOKUP, and TAB1 provide a capability for linear interpolation and extrapolation of table data expressed as a function of as many as three independent variables.

For arguments outside the range of the input data, three options are available:

(1) stop lookup procedure and return error code with function value equal to zero

(2) set function value to data value corresponding to the independent variable value within the input data range which is nearest in value to the desired argument

(3) linearly extrapolate from last available table values to approximate function value at desired argument

The table lookup package consists of one FUNCTION subprogram and two SUBROUTINE subprograms. Four COMMON blocks are required in the calling program to provide for convenient data transfer. The procedures for involving the proper routines and initializing the COMMON blocks are described in the following.

COMMON Block Initialization

COMMON/TABLE/NUMPTS(m) - provides storage for the input table data. NUMPTS is the array where the data are stored and must be dimensioned large enough to hold all tables. An approximation to this dimension may be made as follows:

$$m = \sum_{i=1}^{N} ((X_i + 1)\ (Y_i + 1) + 3)$$

where N is the total number of tables
X_i is the number of first argument values in table i
Y_i is the number of second argument values in table i

Note: a separate table is required for each value of z (third argument) as will be explained later.

COMMON/XTRAP/METH, IACT (3, n) - contains information concerning type of search to be used and action to be taken when argument is outside the range of the table. METH must be initialized in the calling program.

METH = 1: standard order search each time lookup is required

METH = 2: memory search; i.e., search begins at point in table where last search was completed

METH = 3: binary search (can be more efficient)

The second dimension of IACT, n, must be greater than or equal to the number of tables being used.

COMMON/FIND/IPREV (3, n) - stores array data used during interpolation. Here again, n is greater than or equal to the number of tables being used.

COMMON/TABOUT/NTBL, ISEQ - provides information in the event of a data error.

NTBL is the number of the table currently being read.

ISEQ is the sequence number of the card being read.

Data Input

The input data is read from punched cards which have been prepared using the format of Figure 23. Nomenclature pertaining to Figure 23 and certain table data restrictions are given below.

Nomenclature and Restrictions

K	table number (positive integer starting at 1 and increasing by 1 for each successive table)
L1	number of tabular values of the first independent variable (x)
L2	number of tabular values of the second independent variable (y) (= 1 for function of one variable)
Seq. No.	sequence number of card within a table beginning with 0 for the first card, 1 for the second, etc.
z	value of the third independent variable (if used). A separate table is necessary for each value of z.
I1	integer signifying the type of action desired if argument #1(x) goes off table during interpolation = 0: stop lookup procedure and return error code = 1: hold end value of argument #1 = 2: linearly extrapolate on argument #1
I2	same as above but for argument #2
I3	same as above but for argument #3

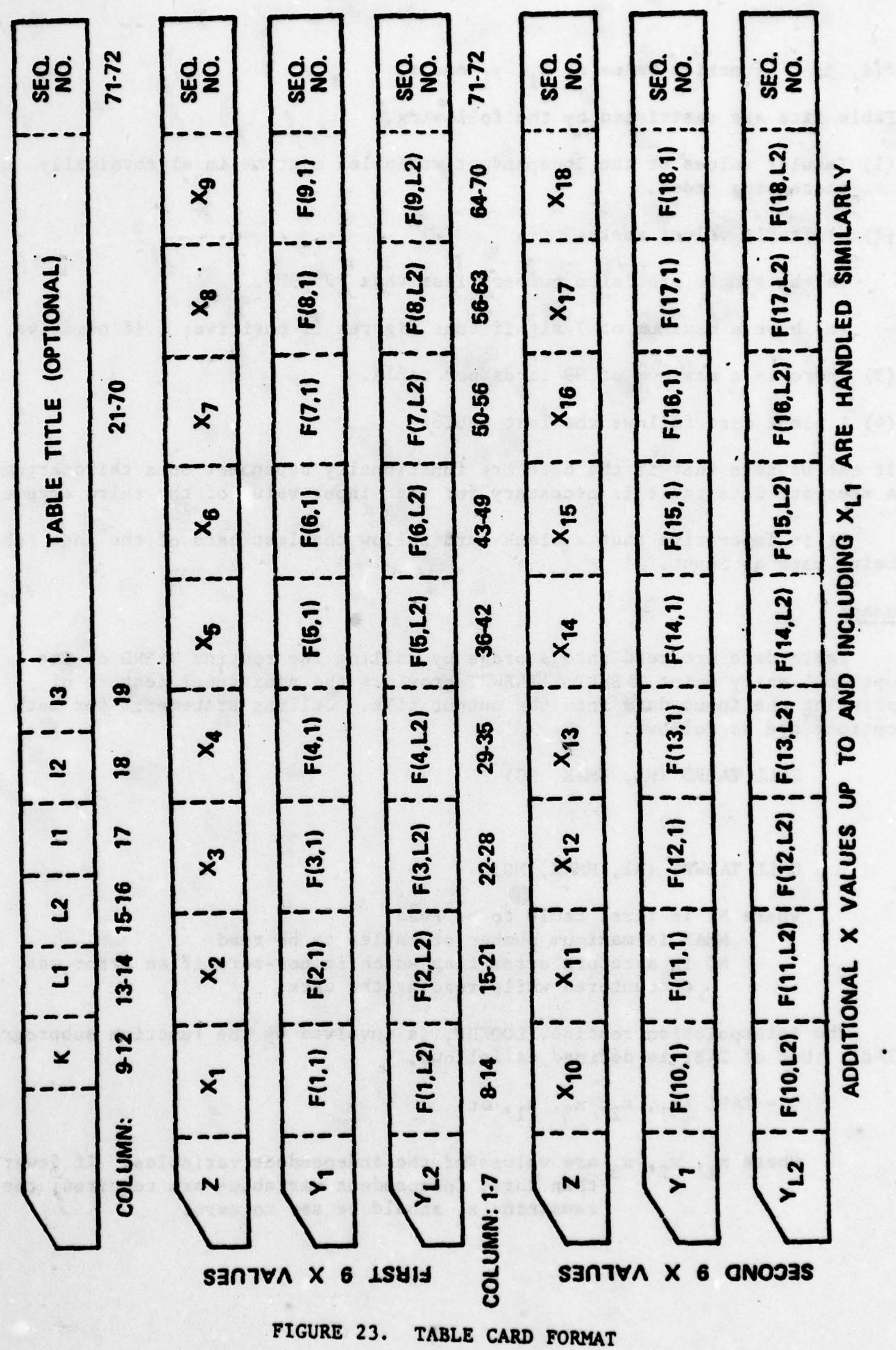


FIGURE 23. TABLE CARD FORMAT

f(i, j) function value at x_i, y_i and z

Table data are restricted by the following.

(1) Tabular values of the independent variables must be in algebraically increasing order.

(2) All table values must:

(a) be single precision numbers less than 99999E9.

(b) have a maximum of 7 significant figures if positive; 6 if negative.

(3) There is a maximum of 99 cards per table.

(4) A blank card follows the last table.

It can be seen that if the data are functionally dependent on a third argument, a separate data table is necessary for each input value of the third argument.

It is imperative that a blank card follow the last card of the last table being used as input.

Usage

Table data are read into storage by calling the routine TABRD or its optional entry point TABWRT. TABWRT provides the additional feature of printing the input data onto the output file. Calling statements for both options are as follows.

```
CALL TABRD (N1, NMAX, NG)

        or

CALL TABWRT (N1, NMAX, NG)
```

where N1 is first table to be read
NMAX is maximum number of tables to be read
NG is a return error flag which is non-zero if an error was encountered while reading the data.

The interpolation routine, LOOKUP, is involved by the function subprogram TAB1. Use of TAB1 is defined as follows.

$$y = TAB1\ (x_1, x_2, x_3, n_1, \Delta n)$$

where x_1, x_2, x_3 are values of the independent variables. If fewer than three independent variables are required, the remaining x_i should be set to zero.

n is the number of the table required for the lookup. If a three argument interpolation is being performed, n is the number of the first required table

Δn is the total number of tables required in the case of a triple argument lookup

y is the resulting function value

If one or more of the arguments is outside the range of the table data, the previously indicated action is taken and an informative message is printed on the output file.

This page intentionally left blank.

APPENDIX E

COMMON VARIABLE DEFINITIONS

This page intentionally left blank.

APPENDIX E
COMMON VARIABLE DEFINITIONS

A Common

NUMBER	NAME	COMPUTED UNITS	DEFINITION
1	PHI	deg	aircraft bank angle
2	THET	deg	aircraft pitch angle
3	PSI	deg	aircraft heading from north
4	PHIR	rad	aircraft bank angle
5	THETR	rad	aircraft pitch angle
6	PSIR	rad	aircraft heading
7	PHID	rad/sec	rate of change of bank
8	THED	rad/sec	rate of change of pitch
9	PSID	rad/sec	rate of change of heading
10	SPHI	-	sine of PHIR
11	CPHI	-	cosine of PHIR
12	STHI	-	sine of THETR
13	CTHT	-	cosine of THETR
14	SPSI	-	sine of PSIR
15	CPSI	-	cosine of PSIR
16	T11	-	euler transformation
17	T21	-	matrix elements
18	T31	-	matrix elements
19	T12	-	matrix elements
20	T22	-	matrix elements
21	T32	-	matrix elements
22	T13	-	matrix elements
23	T23	-	matrix elements
24	T33	-	euler transform coefficient
25	ALFA	deg	aircraft angle of attack
26	BETA	deg	aircraft sideslip angle
27	ALFAR	rad	aircraft angle of attack
28	BETAR	rad	aircraft sideslip angle
29	ALFD	rad/sec	rate of change of ALFAR
30	BETD	rad/sec	rate of change of BETAR
31	SALPH	-	sine of ALFAR
32	CALPH	-	cosine of ALFAR
33	SBETA	-	sine of BETAR
34	CBETA	-	cosine of BETAR
35	GAMV	rad	vertical flight path angle
36	GAMH	rad	horizontal flight path angle
37	PB	rad/sec	aircraft roll rate, body axes
38	QB	rad/sec	aircraft pitch rate, body axes
39	RB	rad/sec	aircraft yaw rate, body axes
40	PL	rad/sec	inertial roll rate due to earth curvature
41	QL	rad/sec	inertial pitch rate due to earth curvature
42	RL	rad/sec	inertial yaw rate due to earth curvature

A Common (Cont'd)

NUMBER	NAME	COMPUTED UNITS	DEFINITION
43	PLB	rad/sec	body axis roll rate due to curvature
44	QLB	rad/sec	body axis pitch rate due to curvature
45	RLB	rad/sec	body axis yaw rate due to curvature
46	PT	rad/sec	total roll rate
47	QT	rad/sec	total pitch rate
48	RT	rad/sec	total yaw rate
49	PBWN	rad/sec	total roll rate WRT wind
50	QBWN	rad/sec	pitch rate WRT wind
51	RBWN	rad/sec	yaw rate WRT wind
52	PTURB	rad/sec	apparent roll rate due to turbulence
53	QTURB	rad/sec	apparent pitch rate due to turbulence
54	RTURB	rad/sec	apparent yaw rate due to turbulence
55	PBD	rad/sec^2	roll acceleration
56	QBD	rad/sec^2	pitch acceleration
57	RBD	rad/sec^2	yaw acceleration
58	UB	ft/sec	body axis - x component of airspeed
59	VB	ft/sec	body axis - y component of airspeed
60	WB	ft/sec	body axis - z component of airspeed
61	UTURB	ft/sec	x - body axis turbulence
62	VTURB	ft/sec	y - body axis turbulence
63	WTURB	ft/sec	z - body axis turbulence
64	VN	ft/sec	aircraft north inertial velocity
65	VE	ft/sec	aircraft east inertial velocity
66	VD	ft/sec	aircraft inertial down velocity
67	VEE	ft/sec	aircraft east velocity relative to ground
68	VT	ft/sec	aircraft total velocity
69	VG	ft/sec	ground velocity
70	VRW	ft/sec	airspeed
71	RMACH XMACH	-	Mach number
72	VNR	ft/sec	relative wind north
73	VER	ft/sec	relative wind east
74	VDR	ft/sec	relative wind down
75	VEQ	kts	equivalent airspeed
76	VNW	ft/sec	north wind component
77	VEW	ft/sec	east wind component
78	VDW	ft/sec	vertical wind component
79	-	-	-
80	ALTD	ft/sec	aircraft sink/climb rate, + up
81	XLOND	rad/sec	rate of change of aircraft longitude
82	XLATD	rad/sec	rate of change of aircraft latitude
83	ALT	ft	aircraft altitude above sea level
84	XLON	rad	aircraft longitude
85	XLAT	rad	aircraft latitude
86	SLAT	-	sine of latitude
87	CLAT	-	cosine of latitude

A Common (Cont'd)

NUMBER	NAME	COMPUTED UNITS	DEFINITION
88	VND	ft/sec^2	aircraft north acceleration
89	VED	ft/sec^2	aircraft east acceleration
90	VDD	ft/sec^2	aircraft vertical acceleration
91	AX	ft/sec^2	x - body axis acceleration at cg
92	AY	ft/sec^2	y - body axis acceleration at cg
93	AZ	ft/sec^2	z - body axis acceleration at cg
94	AXP	ft/sec^2	body axis x acceleration at pilot station
95	AYP	ft/sec^2	body axis y acceleration at pilot station
96	AZP	ft/sec^2	body axis z acceleration at pilot station
97	G	ft/sec^2	acceleration of gravity
98	-	-	-
99	-	-	-
100	-	-	-
101	VCAL	kts	calibrated airspeed
102	-	-	-
103	XPR	ft	ship axis pilot x position
104	YPR	ft	ship axis pilot y position
105	HPR	ft	pilot eye altitude above sea level
106	DNR	ft	relative position of aircraft cg north of reference
107	DER	ft	relative position of aircraft cg east of reference
108	RR	ft	height of runway above center of earth
109	RTV	ft	height of aircraft above center of earth
110	COURSE	rad	ship heading
111	XLATR	rad	latitude of reference point
112	XLONR	rad	longitude of reference point
113	CLATR	-	cosine of latitude
114	STHETR	-	sine of aircraft attitude
115	CTHETR	-	cosine of aircraft attitude
116	XIXX	$slug\text{-}ft^2$	aircraft roll inertia in body axes
117	XIYY	$slug\text{-}ft^2$	aircraft pitch inertia
118	XIZZ	$slug\text{-}ft^2$	aircraft yaw inertia
119	XIXZ	$slug\text{-}ft^2$	aircraft cross inertia
120	XMC(1)	-	inertial coefficient
121	XMC(2)	-	inertial coefficient
122	XMC(3)	$1/slug\text{-}ft^2$	inertial coefficient
123	XMC(4)	$1/slug\text{-}ft^2$	inertial coefficient
124	XMC(5)	-	inertial coefficient
125	XMC(6)	-	inertial coefficient
126	XMC(7)	$1/slug\text{-}ft^2$	inertial coefficient
127	XMC(8)	-	inertial coefficient
128	XMC(9)	-	inertial coefficient
129	XMC(10)	$1/slug\text{-}ft^2$	inertial coefficient
130	XMASS	slug	aircraft mass
131	CL	-	lift coefficient
132	CD	-	drag coefficient

A Common (Cont'd)

NUMBER	NAME	COMPUTED UNITS	DEFINITION
133	CX	-	aircraft x axis aero force coefficient
134	CY	-	y axis aero force coefficient
135	CZ	-	z axis aero force coefficient
136	FAX	lb	aero x axis force
137	FAY	lb	y axis aero force
138	FAZ	lb	z axis aero force
139	FEX	lb	x axis engine force
140	FEY	lb	y axis engine force
141	FEZ	lb	z axis engine force
142	FGX	lb	x axis landing gear force
143	FGY	lb	y axis landing gear force
144	FGZ	lb	z axis landing gear force
145	FTX	lb	x axis total force
146	FTY	lb	y axis total force
147	FTZ	lb	z axis total force
148	FN	lb	total aircraft force north
149	FE	lb	total aircraft force east
150	FD	lb	total aircraft force down
151	FG	lb	gravity force
152	CLL	-	rolling moment coefficient
153	CLM	-	pitching moment coefficient
154	CLN	-	yawing moment coefficient
155	TAL	ft-lb	aerodynamic rolling moment
156	TAM	ft-lb	aerodynamic pitching moment
157	TAN	ft-lb	aerodynamic yawing moment
158	TEL	ft-lb	engine rolling moment
159	TEM	ft-lb	engine pitching moment
160	TEN	ft-lb	engine yawing moment
161	TGL	ft-lb	landing gear rolling moment
162	TGM	ft-lb	gear pitching moment
163	TGN	ft-lb	gear yawing moment
164	TTL	ft-lb	total rolling moment
165	TTM	ft-lb	total pitching moment
166	TTN	ft-lb	total yawing moment
167	DT1	sec	loop1 frame time
168	DT2	sec	loop2 frame time
169	DT3	sec	loop3 frame time
170	HR	ft	runway height above sea level
171	-	-	
172	-	-	
173	-	-	
174	XCG	ft	aircraft x location in ship coordinates
175	YCG	ft	aircraft y location
176	HCG	ft	aircraft height above runway
177	WAIT	lb	aircraft weight
178	QBAR	lb/ft^2	dynamic pressure

A Common (Cont'd)

NUMBER	NAME	COMPUTED UNITS	DEFINITION
179	QBARC	lb/ft^2	stagnation - static pressure
180	AREA	ft^2	wing area
181	SPAN	ft	wing span
182	CHORD	ft	mean aerodynamic chord
183	RHO	$slug/ft^3$	air density
184	DXPA(1)	ft	nose gear x station
185	DXPA(2)	ft	main gear x station
186	DXPA(3)	ft	right wing gear x station
187	DXPA(4)	ft	left wing gear x station
188	DXPA(5)	ft	pilot x station
189	DXPA(6)	ft	tail x station
190	DXPA(7)	ft	ram drag x station
191	DXPA(8)	ft	front nozzle x station
192	DXPA(9)	ft	rear nozzle x location
193	DXPA(10)	ft	front pitch jet x station
194	DXPA(11)	ft	rear pitch jet x station
195	DXPA(12)	ft	yaw jet x station
196	DXPA(13)	ft	right wing jet x station
197	DXPA(14)	ft	left wing jet x station
198	DXCG	ft	cg reference x station
199			
200	VX	ft/sec	aircraft x axis inertial velocity
201	VY	ft/sec	aircraft y axis inertial velocity
202			
203			
204			
205			
206			
207			
208			
209	SOUND	ft/sec	speed of sound
210	DYPA(1)	ft	nose gear y station
211	DYPA(2)	ft	main gear y station
212	DYPA(3)	ft	right wing gear y station
213	DYPA(4)	ft	left wing gear y position
214	DYPA(5)	ft	pilot y station
215	DYPA(6)	ft	tail y station
216	DYPA(7)	ft	ram drag y station
217	DYPA(8)	ft	front nozzle y station
218	DYPA(9)	ft	rear nozzle y location
219	DYPA(10)	ft	front pitch jet y station
220	DYPA(11)	ft	rear pitch jet y station
221	DYPA(12)	ft	yaw jet y station
222	DYPA(13)	ft	right wing jet y location
223	DYPA(14)	ft	left wing jet y location
224	DYPA(15)	ft	aircraft lateral cg position

A Common (Cont'd)

NUMBER	NAME	COMPUTED UNITS	DEFINITION
225	XP	ft	pilot x position WRT cg
226	YP	ft	pilot y position WRT cg
227	ZP	ft	pilot z position WRT cg
228	-	-	
229	VDIC	ft/sec	trim vertical velocity
230	PHIIC	deg	trim bank angle
231	THETIC	deg	trim pitch attitude
232	PSIIC	deg	trim heading angle
233	GAMVIC	deg	initial vertical path angle
234	GAMHIC	deg	horizontal path angle
235	PBIC	rad/sec	initial roll rate
236	QBIC	rad/sec	initial pitch rate
237	RBIC	rad/sec	initial yaw rate
238	VEQIC	kts	trim equivalent airspeed
239	XIC	ft	initial aircraft x location
240	YIC	ft	initial aircraft y location
241	HIC	ft	initial altitude
242	WAITIC	lb	initial aircraft weight
243	XIXXIC	slug-ft^2	initial roll inertia
244	XIYYIC	slug-ft^2	initial pitch inertia
245	XIZZIC	slug-ft^2	initial yaw inertia
246	XIXZIC	slug-ft^2	initial cross-product inertia
247	WFUELIC	lb	initial fuel weight
248	WWATIC	lb	initial water weight
249	WFUEL	lb	fuel weight
250	WWAT	lb	water weight
251	WAITO	lb	initial empty weight
252	WSTORE	lb	store weight
253	-		
254	DXCGO	ft	x location of mac leading edge
255	DZCGO	ft	z location of aero data reference chord
256	CG	percent	aircraft center of gravity WRT mac
257	DZPA(1)	ft	nose gear z location
258	DZPA(2)	ft	main gear z station
259	DZPA(3)	ft	right wing gear z station
260	DZPA(4)	ft	left wing gear z station
261	DZPA(5)	ft	pilot z position
262	DZPA(6)	ft	tail z position
263	DZPA(7)	ft	ram drag z position
264	DZPA(8)	ft	front nozzle z position
265	DZPA(9)	ft	rear nozzle z position
266	DZPA(10)	ft	front nozzle pitch jet z position
267	DZPA(11)	ft	rear pitch jet z position
268	DZPA(12)	ft	yaw jet z position
269	DZPA(13)	ft	right wing jet z position
270	DZPA(14)	ft	left wing jet z position

A Common (Cont'd)

NUMBER	NAME	COMPUTED UNITS	DEFINITION
271	DZPA(15)	ft	cg - cg reference z position
272	HDECK(1)	ft	deck height beneath nose gear
273	HDECK(2)	ft	deck height beneath main gear
274	HDECK(3)	ft	deck height beneath right wing gear
275	HDECK(4)	ft	deck height beneath left wing gear
276	HDECK(5)	ft	deck height beneath tail
277	-		
278	-		
279	XIG(1)	ft	inertial x location of nose gear
280	XIG(2)	ft	inertial x location of main gear
281	XIG(3)	ft	
282	XIG(4)	ft	
283	XIG(5)	ft	
284	XIG(6)	ft	
285	YIG(1)	ft	inertial y position of landing gear
286	YIG(2)	ft	
287	YIG(3)	ft	
288	YIG(4)	ft	
289	YIG(5)	-	
290	YIG(6)	-	
291	HIG(1)	ft	altitude of nose gear above sea level
292	HIG(2)	ft	altitude of main gear
293	HIG(3)	ft	altitude of right wing gear
294	HIG(4)	ft	altitude of left wing gear
295	HIG(5)	ft	-
296	HIG(6)	ft	-
297	-		
298	-		
299	-		
300	-		
301	-		
302	-		
303	TIME	sec	run time
304	RTIME	sec	maximum run time
305-331	-	-	-
332	TR	-	total temperature ratio
333	PR	-	total pressure ratio
334	-	-	-
335	DTIM1	sec	print time increment
336	TI	sec	print time interval switch point
337-357	-	-	-
358	D2R	rad/deg	converts radians to degrees
359	R2D	deg/rad	converts degrees to radians
360-363	-	-	-
364	RHOZ	slug/ft^3	reference air density
365	HRHOZ	ft	reference altitude

A Common (Cont'd)

NUMBER	NAME	COMPUTED UNITS	DEFINITION
366	TAMB	°k	ambient temperature
367	PAMB	lb/ft^2	ambient pressure
368	TTOT	°k	total temperature
369	PTOT	lb/ft^2	total pressure
370	DELAT	°k	temperature variation from standard at sea level
371	SQTEMPR	-	square root of temperature ratio
372-374	-	-	-
375	XMCC1	1/sec	inertial coefficient
376	XMCC2	1/sec	inertial coefficient
377	XMCC3	1/sec	inertial coefficient
378	XMCC4	1/sec	inertial coefficient
379	XMCC5	1/sec	inertial coefficient
380	XMCC6	1/sec	inertial coefficient
381	XMCC7	1/sec	inertial coefficient
382	EXMX	ft-lb-sec	rotor angular momentum
383	EXMY	ft-lb-sec	rotor angular momentum
384	EXMZ	ft-lb-sec	rotor angular momentum
385-390	-	-	-
391	STATE(1)	-	trim state variable
392	STATE(2)	-	trim state variable
393	STATE(3)	-	trim state variable
394	STATE(4)	-	trim state variable
395	STATE(5)	-	trim state variable
396	STATE(6)	-	trim state variable
397	CONT(1)	-	trim control variable
398	CONT(2)	-	trim control variable
399	CONT(3)	-	trim control variable
400	CONT(4)	-	trim control variable
401	CONT(5)	-	trim control variable
402	CONT(6)	-	trim control variable
403-405	-	-	-
406	UBIC	ft/sec	trim u velocity
407	VBIC	ft/sec	trim v velocity
408	WBIC	ft/sec	trim w velocity
409-412	-	-	-
413	UBD	ft/sec^2	rate of change of UB
414	VBD	ft/sec^2	rate of change of VB
415	WBD	ft/sec^2	rate of change of WB
416	VTWN	ft/sec	total north wind
417	VTWE	ft/sec	total east wind
418	VTWD	ft/sec	total vertical wind
419	VNTURB	ft/sec	north component of turbulence
420	VETURB	ft/sec	east component of turbulence
421	VDTURB	ft/sec	vertical turbulence
422-432	-	-	-

A Common (Cont'd)

NUMBER	NAME	COMPUTED UNITS	DEFINITION
433	VDKN(1)	ft/sec	deck north velocity at nose gear location
434	VDKN(1)	ft/sec	deck north velocity at main gear
435	VDKN(1)	ft/sec	deck north velocity at right wing gear
436	VDKN(1)	ft/sec	deck north velocity at left wing gear
437	-	-	
438	-	-	
439	-	-	.
440	VDKE(1)	ft/sec	deck east velocity at nose gear location
441	VDKE(2)	ft/sec	deck east velocity at main gear
442	VDKE(3)	ft/sec	deck east velocity at right wing gear
443	VDKE(4)	ft/sec	deck east velocity at left wing gear
444	VDKE(5)	-	-
445	-		
446	-		
447	HDTDKE(1)	ft/sec	deck vertical velocity at cg
448	HDTDKE(2)	ft/sec	deck vertical velocity at nose gear
449	HDTDKE(3)	ft/sec	deck vertical velocity at main gear
450	HDTDKE(4)	ft/sec	deck vertical velocity at right wing gear
451	HDTDKE(5)	ft/sec	deck vertical velocity at left wing gear
452-453	-	-	-
454	VR	ft/sec	relative aircraft deck velocity
455	FGXM	lb	maximum braking force
456-500	-	-	-

B Common

NUMBER	NAME	COMPUTED UNITS	DEFINITION
1	STAB	deg	stabilizer deflection
2	AIL	deg	aileron deflection
3	RUD	deg	rudder deflection
4	-	-	-
5	STAB1	deg	pilot commanded stabilizer
6	PSAS	deg	pitch SAS command
7	RLONSTK	in	longitudinal stick
8	RLNSTKO	in	trim longitudinal stick
9	RLATSTK	in	lateral stick
10	RLTSTKO	in	trim lateral stick
11	RUDPED	in	rudder pedal
12	YSAS	deg	yaw SAS position
13	RSAS	deg	roll SAS command
14	THROT	deg	throttle position
15	THROTIC	deg	trim throttle
16	THTNC	deg	engine nozzle angle command
17	THTN	deg	engine nozzle angle
18	THTNIC	deg	trim nozzle angle
19	RN1	percent	engine RPM
20	TOUT	lb	engine gross thrust
21	RPEDO	in	trim rudder pedal
22	THRTMN	-	minimum trim throttle
23	THRTMX	-	maximum trim throttle
24	THTMN	deg	minimum trim attitude
25	THTMX	deg	maximum trim attitude
26	PHIMN	deg	minimum trim bank angle
27	PHIMX	deg	maximum trim bank angle
28	PEDMN	in	minimum trim pedal
29	PEDMX	in	maximum trim pedal
30	VS	ft/sec	ship speed
31-32	-	-	-
33	THTNMN	deg	minimum trim nozzle angle
34	THTNMX	deg	maximum trim nozzle angle
35	VNWBIC	ft/sec	north component of wind
36	VEWBIC	ft/sec	east component of wind
37	BETAIC	deg	trim sideslip
38	STKMIN	in	minimum trim longitudinal stick
39	STKMAX	in	maximum trim longitudinal stick
40	RLSMIN	in	minimum lateral trim stick
41	RLSMAX	in	maximum lateral stick
42	ALFAIC	deg	trim angle of attack
43	-	-	-
44	TFPJ	lb	front pitch jet thrust
45	TRPJ	lb	rear pitch jet thrust
46	TYAWJ	lb	yaw jet thrust
47	TRRJ	lb	right roll jet thrust

B Common (Cont'd)

NUMBER	NAME	COMPUTED UNITS	DEFINITION
48	TLRJ	lb	left roll jet thrust
49	TFPJ1	lb	maximum front pitch jet thrust
50	TRPJ1	lb	rear pitch jet reference thrust
51	TRJ1	lb	roll jet reference thrust
52	TYAWJ1	lb	yaw jet reference thrust
53	WWDOT1	lb/sec	engine water flow rate
54	-	-	-
55	-	-	-
56	TAUENG	sec	engine time constant
57	RDRAG	lb	ram drag
58	TFN	lb	front nozzle gross thrust
59	TRN	lb	rear nozzle gross thrust
60	VJRCS	ft/sec	RCS jet velocity
61	DGE	-	% thrust loss due to suckdown
62	ALT1	ft	ground effect altitude limit
63	GEF1	-	suckdown coefficient
64	GEF2	1/ft	suckdown coefficient
65	ALTGE	ft	suckdown coefficient
66	FCR	lb/sec/%	fuel consumption rate
67	ALRJ	in^2	left roll jet area
68	AFPJ	in^2	front pitch jet area
69	ARPJ	in^2	rear pitch jet area
70	ARRJ	in^2	right roll jet area
71	AYAWJ	in^2	yaw jet area
72	CSPLAYF	-	cosine of front nozzle splay angle
73	CSPLAYR	-	cosine of rear nozzle splay angle
74	CRJ	-	cosine of roll jet angle
75	CFPJA	-	cosine of front pitch jet angle
76	CRPJA	-	cosine of rear pitch jet angle
77	SFPJA	-	sine of front pitch jet angle
78	SRPJA	-	sine of rear pitch jet angle
79	SRJ	-	sine roll jet angle
80	RLAM1	rad	component of deck tilt along aircraft x axis
81	RLAM2	rad	component of deck tilt along aircraft y axis
82	PHIS	rad	ship roll angle
83	THETS	rad	ship pitch angle
84	PSIS	rad	ship yaw angle
85	THETSD	rad/sec	ship pitch rate
86	PHISD	rad/sec	ship roll rate
87	PSISD	rad/sec	ship yaw rate
88	HEAVE	ft	ship heave at cg
89	HEAVED	ft/sec	ship heave velocity at cg
90	SURGE	ft	ship fore-aft motion
91	SURGED	ft/sec	ship fore-aft transient velocity

B Common (Cont'd)

NUMBER	NAME	COMPUTED UNITS	DEFINITION
92	SWAY	ft	ship transient lateral motion
93	SWAYD	ft/sec	ship lateral velocity
94	X1S	ft	aft deck limit
95	X2S	ft	forward deck limit
96	Y1S	ft	port deck limit
97	Y2S	ft	starboard deck limit
98	X3S	ft	bow location
99	H3S	ft	superstructure height
100	H1S	ft	deck mean height
101	XTD	ft	longitudinal ideal touchdown point
102	YTD	ft	lateral ideal touchdown point
103	HIPS	ft	ship roll/pitch height
104	HMAG	ft	ship heave magnitude
105	PHIROL	rad	phase of ship roll
106	PHIHV	rad	phase of ship roll
107	OMEGH	rad/sec	frequency of heave motion
108	OMEGR	rad/sec	frequency of ship roll
109	ROLLM	rad	magnitude of ship roll
110	PITCHM	rad	ship pitch magnitude
111	OMEGP	rad/sec	ship pitch frequency
112	PHIP	rad	ship pitch phase
113	YAWM	rad	ship yaw magnitude
114	OMEGY	rad/sec	frequency of ship yaw
115	PHIY	rad	phase of ship yaw
116	SURGM	ft	magnitude of ship surge
117	OMEGSRG	rad/sec	frequency of ship surge
118	PHISRG	rad	phase of ship surge
119	SWAYM	ft	maximum ship sway
120	OMEGSWY	rad/sec	frequency of ship sway
121	PHISWY	rad	phase of ship roll
122	VJR	-	jet nozzle velocity ratio
123	CDQ	1/rad	drag coefficient due to pitch rate
124	CLADOT	1/rad	lift coefficient due to rate of change of angle of attack
125	CLP	1/rad	roll moment coefficient due to roll rate
126	CMQ	1/rad	pitching moment coefficient due to pitch rate
127	CYDR	1/deg	sideforce coefficient due to rudder
128	CYR	1/rad	sideforce coefficient due to yaw rate
129	CLDA	1/deg	roll moment coefficient due to aileron
130	CLQ	1/rad	lift coefficient due to pitch rate
131	CMADT	1/rad	pitching moment coefficient due to rate of change of alfa
132	CNDR	1/deg	yawing moment coefficient due to rudder
133	CNR	1/rad	yaw moment coefficient due to yaw rate
134	CYDA	1/deg	sideforce coefficient due to aileron

B Common (Cont'd)

NUMBER	NAME	COMPUTED UNITS	DEFINITION
135	CYP	1/rad	sideforce coefficient due to roll rate
136-139	-	-	-
140	HE	ft	height of landing gear above touchdown point
141	XE	ft	x distance from aircraft to touchdown point
142	YE	ft	y distance from aircraft to touchdown point
143	ALTCOM	ft	commanded altitude
144	ALTDCOM	ft/sec	commanded climb rate
145	YCOM	ft	commanded lateral position
146	XCOM	ft	commanded x position
147	THETCOM	deg	pitch command
148	PHICOM	deg	bank command
149	VXCOM	ft/sec	commanded airspeed
150	PSICOM	deg	heading command
151	CMSTR1	-	store pitch coefficient
152	CLSTR3	-	store lift increment
153	UGST	$(ft/sec)^2$	rms u gust
154	VGST	$(ft/sec)^2$	mean square v gust
155	WGST	$(ft/sec)^2$	mean square w gust
156	XFRZ	ft	burble freeze range
157	SWBSW	-	w burble scale factor
158	SUBSW	-	u burble scale factor
159	-	-	-
160	RKQ	deg/rad/sec	SAS pitch rate gain
161	TLEADP	sec	pitch SAS lead time constant
162	PSAS1	deg/sec	pitch SAS feedback term
164	-	-	-
165	RBWSHI	deg/sec	yaw SAS coefficient
166	RBWSH	deg	output of yaw rate washout filter
167	RKDRRB	deg/rad/sec	yaw rate to rudder gain
168	TLAGRB	sec	yaw rate lag constant
169	RKDRAY	deg/ft/sec	yaw SAS gain
170	TLEADAY	sec	yaw SAS time constant
171	TLAGAY	sec	yaw SAS time constant
172	AYFILTI	deg	lateral acceleration filter term
173	AYFILT	deg	lateral acceleration filter output
174	YSAS2	deg	lateral SAS output
175	RINCNT	deg	rudder command due to aileron
176	RKDRDA	deg/deg	rudder interconnect gain
177	YSAS1	deg	SAS input to rudder
178	YSASLIM	deg	SAS rudder limit
179	RPBDA	deg/rad/sec	roll SAS gain
180	TLEADDA	sec	roll SAS time constant
181	TLAGDA	sec	roll SAS time constant

B Common (Cont'd)

NUMBER	NAME	COMPUTED UNITS	DEFINITION
182	PBWSH	deg	output of roll rate washout
183	PWSHI	deg	roll SAS term
184	PSASLIM	deg	pitch SAS limit
185	AILC	deg	aileron command
186	TAUACDA	sec	aileron time constant
187	TAUACHS	sec	elevator time constant
188	RUD2	deg	total RCS yaw command
189	XTDI	ft	inertial x location of touchdown point
190	YTDI	ft	y location of touchdown point
191	TAUALPH	sec	pilot model time constant
192-200	-	-	-

IB Common

NUMBER	NAME	DEFINITION
1	IHOV	hover configuration flag
2	ICON	conventional flight flag
3	ILAND	deck landing switch
4	IZERO	statistical array zero flag
5	IHNDOFF	hands off takeoff switch
6	ITRANS	transition flight flag
7	ISWWAT	water injection switch
8	IFUEL	fuel exhaustion flag
9	ICRASH	crash flag
10	-	-
11	ICOPY	statistics print control
12	ISET	brake application switch
13	-	-
14	ITD	touchdown flag
15	IDA	deck edge crossing flag
16	IONCE	turbulence sequence control
17	IHOLD	altitude hold switch
18	IPITCH	ship pitch induced wind switch
19	IHIT	ground trim flag
20	IWIND	turbulence switch
21	ILTURB	turbulence switch
22	NCHK	control input selector
23	MCHK	input type selector
24	IBFRZ	burble freeze control
25	ILBURB	burble control switch
26	IPDAMP	roll damper switch
27	IQDAMP	pitch damper switch
28	IRDAMP	yaw damper switch
29	IBRAK	brake application switch
30	ITASK	control task switch
31	IGRAD	trim gradient evaluation switch
32-35	-	-
36	NCNT	calcomp plot point count
37	ISTOR2	outboard store flag
39	-	-
40	ITHIS	time response print switch
41	IUPLOT	calcomp plot switch
42	ISTAT	statistics printout switch
43-44	-	-
45	IPITCH	ship pitch induced wind
46-48	-	-
49	J2	trim iteration limit
50	JLAND	landing switch

32 I HTrack Pilot Model - hover Command Switch

IA Common

NUMBER	NAME	DEFINITION
1	IMODE	mode switch
2-5	-	-
6	IFLAT	flat earth switch
7	IFFCI	constant position switch
8-9	-	-
10	ITOUCH(1)	nose gear contact flag
11	ITOUCH(2)	main gear contact flag
12	ITOUCH(3)	right wing gear contact flag
13	ITOUCH(4)	left wing gear contact flag
14	ITOUCH(5)	-
15	IOVDK	over deck switch
16	I2DOF	2 degree of freedom pitch following mode
17-20	-	-
21	IEVAL	trim gradient evaluation flag
22	-	-
23	ITPROG	trim switch
24-60	-	-
61	IDT1	first loop time increment milli-sec
62	IDT2	second loop time increment milli-sec
63	IDT3	third loop time increment milli-sec
64	ICG	trim altitude flag
65	ID	trim flag
66-89	-	-
90-101	ICHNBF(I,J)	printout title number
102	ITRIM	trim switch
103	ITRMP	past value of trim switch
104	IGEAR	landing gear switch
105	NRUN	number of runs
106-109	-	-
110	INDEXT	trim mode switch
111	ITRMCM	trim command switch
112-113	-	-
114	IMACH	airspeed trim switch
115-117	-	-
118	J1	trim iteration count
119-123	-	-
124	ICPRNT	initial condition print switch
125-140	-	-
141	ICOND	constant density flag
142-166	-	-
167	IM	trim state counter
168-178	-	-
179	IREVAL	gradient flag
180	NUSED	loop 3/loop 2 frame time
181-183	-	-
184	IEULR	trim velocity flag
185	IETURB	turbulence coordinate flag

115 IInRef Deck

IA Common (Cont'd)

NUMBER	NAME	DEFINITION
186	-	-
187	ITOMTR	alfa, beta rate flag
188-198	-	-
199	N2	run counter
200	-	-

24 Itape - Ship input Flag

25 IStart Ship input Flag

26 IFile Ship input Flag

27 K1 "

28 JFILMX "

29 JStart "

30-50 NLag - Pilot model Lag index

GA Common

NUMBER	NAME	COMPUTED UNITS	DEFINITION
1	FAC	-	plot scale factor
2-17	-	-	-
18	FREQCK	rad/sec	sinusoidal input frequency
19	T0	sec	input switch time
20	T1	sec	input switch time
21	T2	sec	input switch time
22	RMAG	-	input magnitude
23	DTIM2	sec	printout time interval
24-29	-	-	-
30-35	GA(I)	ft/sec	turbulence scale
36	VERI1	ft	throttle integral feedback limit
37	ALT1	ft	control switch altitude
38	RK1V	deg/ft/sec	throttle gain
39	RK2V	deg/ft	throttle gain
40	-	-	-
41	RKRUD1	in/rad/sec	yaw rate to rudder gain
42	RKRUD2	in/rad	sideslip to rudder gain
43	ALT2	ft	switch altitude
44	RKTHT1	$rad/ft/sec^2$	pitch command gain
45	RKALPH	rad/rad	angle of attack to pitch command gain
46	THET1	rad	minimum pitch attitude command
47	ALT3	ft	altitude command
48	RKALTD	1/sec	altitude rate command gain
49	THETCO	rad	pitch attitude command
50	RKTHT2	rad/ft	pitch command gain
51	RKTHT3	rad/ft/sec	pitch command gain
52	RKTHT4	rad/ft/sec	pitch command gain
53	RKYE	rad/ft	roll command gain
54	RKVE	rad/ft/sec	roll command gain
55	RKPSI	rad/rad	roll command gain
56	RKPHILT	in/rad	lateral stick gain
57	RKPLT	in/rad/sec	lateral stick gain
58	RKTHLN	in/rad	longitudinal stick gain
59	RKQLN	in/rad/sec	longitudinal stick gain
60	RKNOZLN	in/deg	longitudinal stick/nozzle angle gain
61	VRW1	ft/sec	switching speed
62	ALT4	ft	switching altitude
63	THTNC1	deg	nozzle angle command
64	XCG1	ft	switching range
65	VRW2	ft/sec	switching speed
66	VRW3	ft/sec	switching speed
67	VRW4	ft/sec	switching speed
68	ALT5	ft	switching altitude
69	RKTHT5	rad/ft	pitch command gain
70	RKVY1	-	bank command gain
71	RKVX1	rad/ft/sec	pitch command gain

GA Common (Cont'd)

NUMBER	NAME	COMPUTED UNITS	DEFINITION
72	RKVY2	-	bank command gain
73	RKVX2	rad/ft/sec	pitch command gain
74	RKYPHI	rad/ft	bank command gain
75	RKPHIVY	rad/ft/sec	bank command gain
76	RLNTHT2	in/rad	longitudinal stick gain
77	RLNQB2	in/rad/sec	longitudinal stick gain
78	RLTPHI2	in/rad	lateral stick gain
79	RLTPB2	in/rad/sec	lateral stick gain
80	RPR1	in/rad/sec	rudder gain
81	RPPSI1	in/rad	rudder gain
82	RTHROT	deg/ft	throttle gain
83	RTHROT1	deg/ft/sec	throttle gain
84	RTHROT2	deg/ft/sec	throttle gain
85	ALFTRM1	deg	reference angle of attack
86	XCG2	ft	switch range
87	TPITCH	sec	pitch command time limit
88	THET2	rad	pitch command limit
89	RPMMAX	RPM	maximum engine speed
90	THCOR	-	engine thrust correction factor
91	RTHROT3	$deg/ft/sec^2$	throttle gain

This page left intentionally blank.

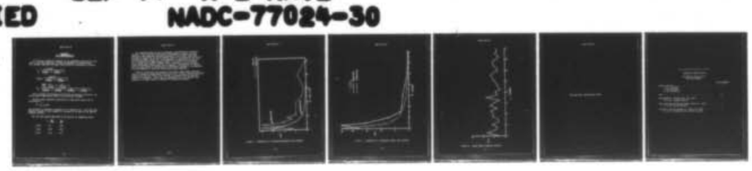
APPENDIX F

GUST CHARACTERISTICS

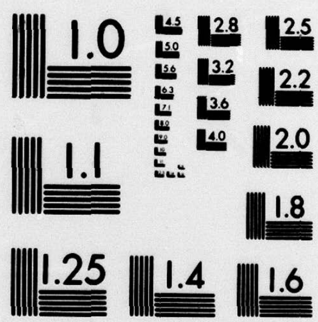
This page left intentionally blank.

AD-A073 587 NAVAL AIR DEVELOPMENT CENTER WARMINSTER PA AIR VEHICL--ETC F/G 1/2
A COMPUTERIZED VSTOL/SMALL PLATFORM LANDING DYNAMICS INVESTIGAT--ETC(U)
SEP 77 R L NAVE
UNCLASSIFIED NADC-77024-30

4 OF 4
AD
A073587



END
DATE
FILMED
10-79
DDC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

APPENDIX F
GUST CHARACTERISTICS

The free air turbulence component of the atmospheric disturbance model used in the VSTOL simulation is based on Appendix B of MILSPEC AR-40. This document specifies the power spectrum for air turbulence as follows:

$$\Phi_{\sim g} = \frac{71.6/Vo^3}{\left(\frac{S}{Vo/100}+1\right)\left(\frac{-S}{Vo/100}+1\right)}$$

$$\Phi_{\Delta ug/uo} = \frac{200/Vo^3}{\left(\frac{S}{Vo/100}+1\right)\left(\frac{-S}{Vo/100}+1\right)}$$

$$\Phi_{S g} = \frac{\left(\frac{940/3}{Vo}\right)\left(\frac{S}{Vo/400}+1\right)\left(\frac{-S}{Vo/400}+1\right)}{\left(\frac{-S}{3Vo/400}+1\right)\left(\frac{S}{3Vo/400}+1\right)\left(\frac{-S}{Vo/1000}+1\right)\left(\frac{S}{Vo/1000}+1\right)}$$

These represent the frequency distribution of vertical, horizontal, and lateral gusts respectively, and Vo is the aircraft airspeed.

The mean square amplitude represented by these power spectra may be calculated as

$$G^2 = \int_{-\infty}^{\infty} s(w)\,dw$$

This integral is tabulated in Appendix IV of reference (f). The factor $1/2\pi$ normally included in this integral is contained in the gain of the spectral transfer function.

The root mean square amplitudes of the spectra are summarized below:

	G - $\frac{ft}{sec}$	$\frac{m}{sec}$
U gust	2.5	.762
V gust	2.39	.728
W gust	1.50	.457

• The simulation model was used to generate a representative turbulence sequence. Then these data were fourier transformed to compute the power spectrum of the simulated turbulence generated by the digital algorithms. As shown in Figures 24 and 25, the general shape of the theoretical and actual distribution agree rather well. The comparison is complicated by the fact that the finite data sample length results in a cutoff of the DC portion of the signal and a rather irregular curve. Therefore, only the general shape and frequency distribution can be compared. Nevertheless, it appears that the model should be satisfactory for simulation purposes if the magnitude is scaled properly to yield the desired mean square amplitude.

Figure 26 illustrates the spectrum of the basic random number generator which is intended to approximate white noise with a constant spectral amplitude. Although the spectrum is not truly uniform, it does not show any definite trends with frequency. Thus, it may be considered flat in an average sense. Therefore, the model should be adequate for simulation purposes.

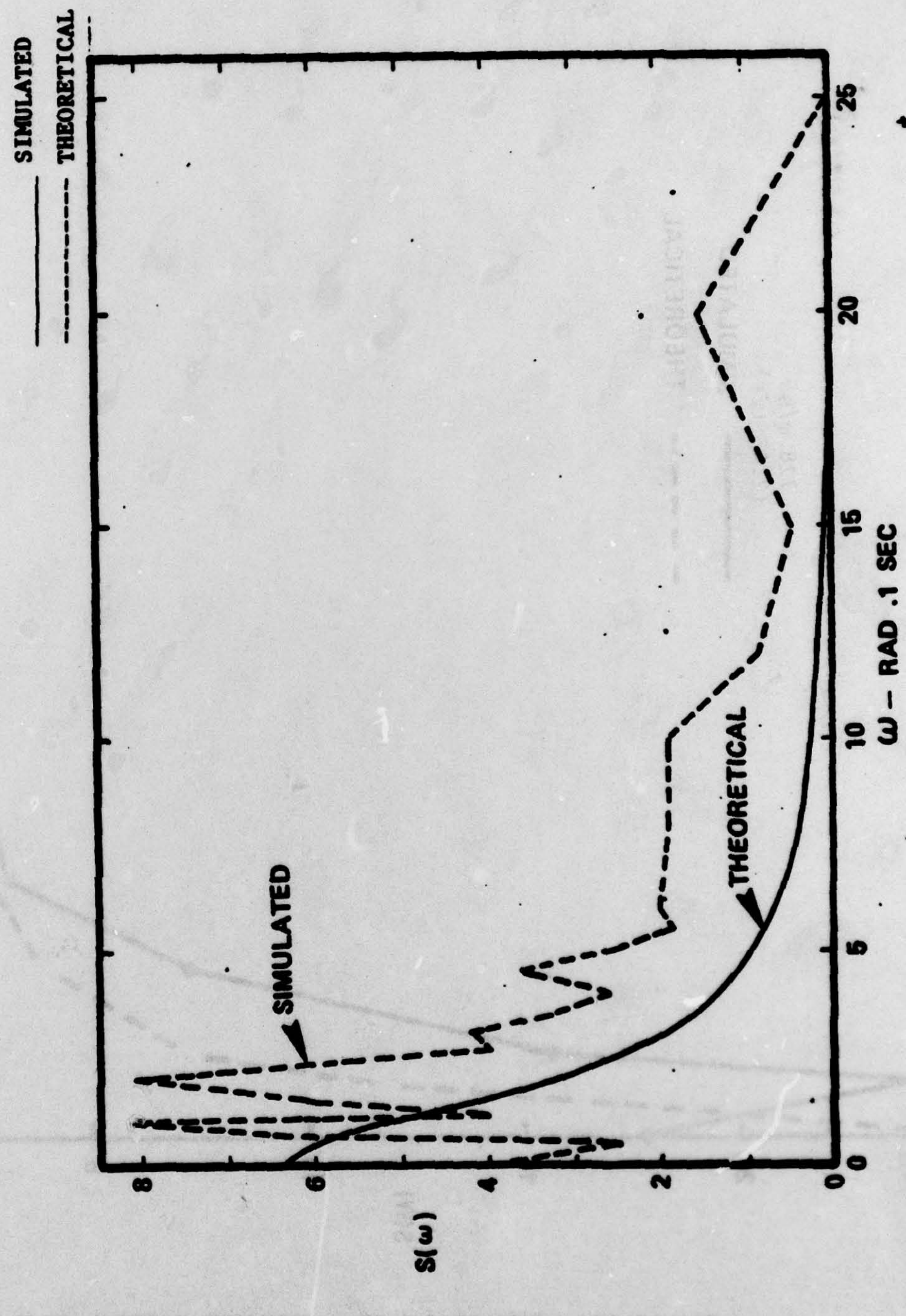


FIGURE 24. THEORETICAL VS. SIMULATED HORIZONTAL GUST SPECTRUM

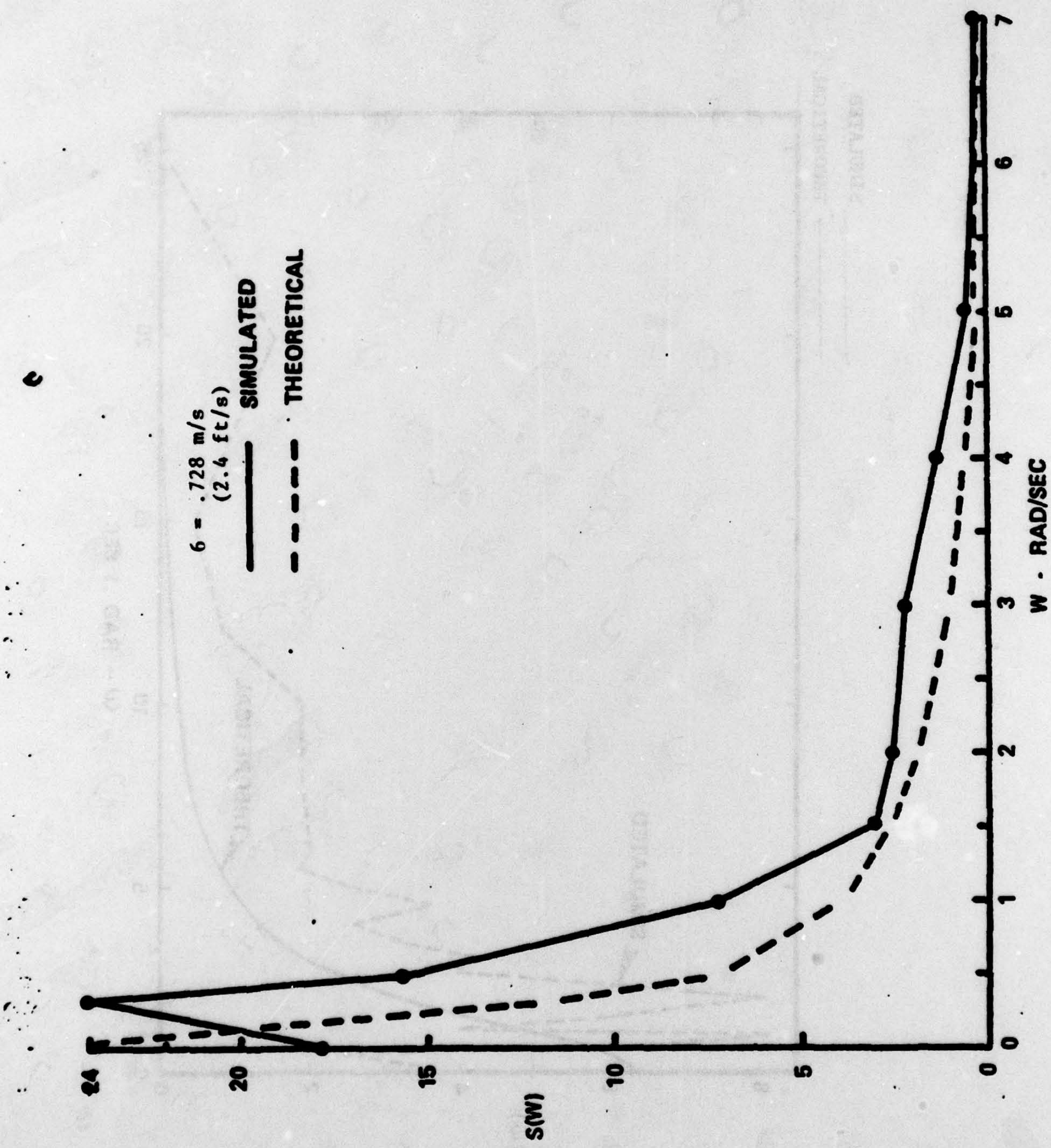


FIGURE 25. THEORETICAL VS. SIMULATED LATERAL GUST SPECTRUM

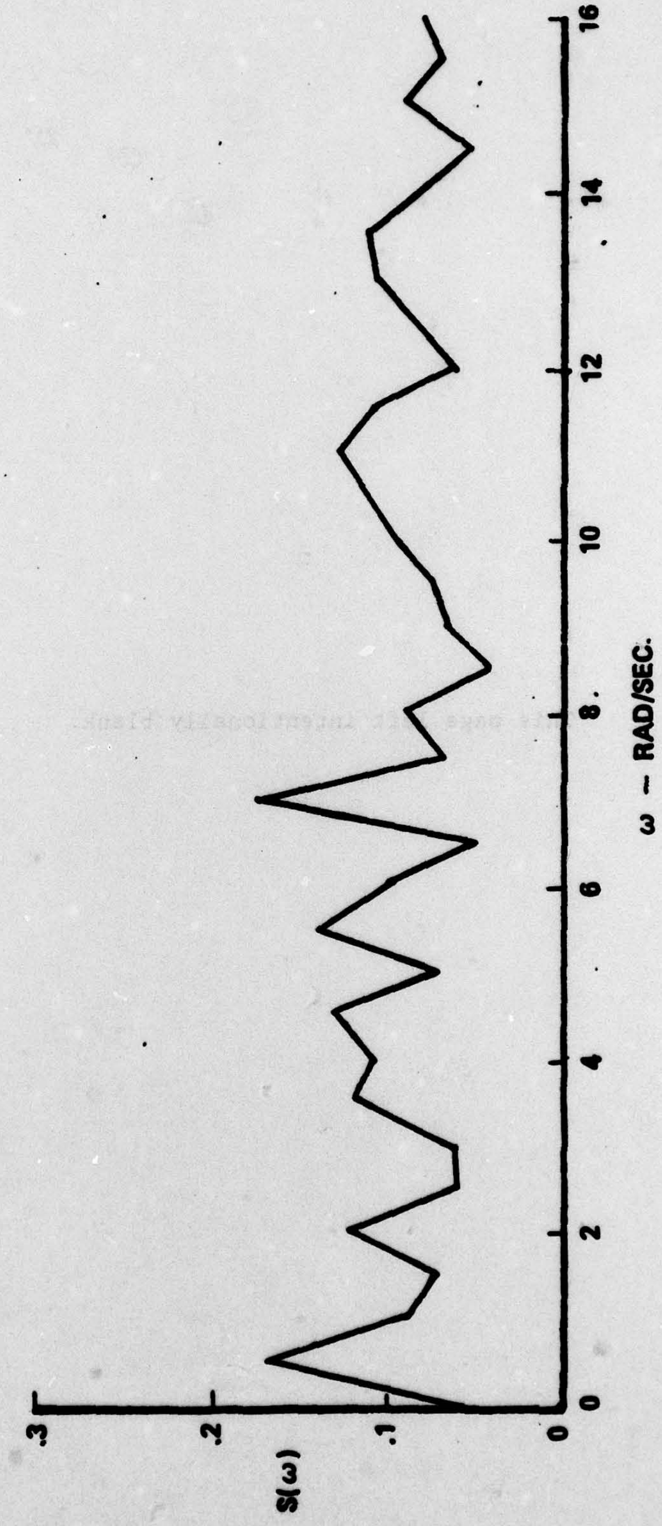


FIGURE 26. RANDOM NUMBER GENERATOR SPECTRUM

This page left intentionally blank.

D I S T R I B U T I O N L I S T

REPORT NO. NADC-77024-30

AIRTASK NO. WF41-421-203
WORK UNIT MX2XX1

	No. of Copies
NAVAIR (AIR-954) . . . . (2 for retention) (3 for AIR-320D) (1 for AIR-530112)	6
DDC	12
NAVAIRTESTCEN, Patuxent River, MD 20670 . . . . (1 for Mr. Anthony Rosetti)	1
NASA Ames Research Center, Moffett Field, CA 94035 . . . . (1 for Mr. Walter McNeill)	1
McDonnell Aircraft Company, St. Louis, MO 63166 . . . . . . (1 for Mr. Douglas Miller, Harrier Project)	1